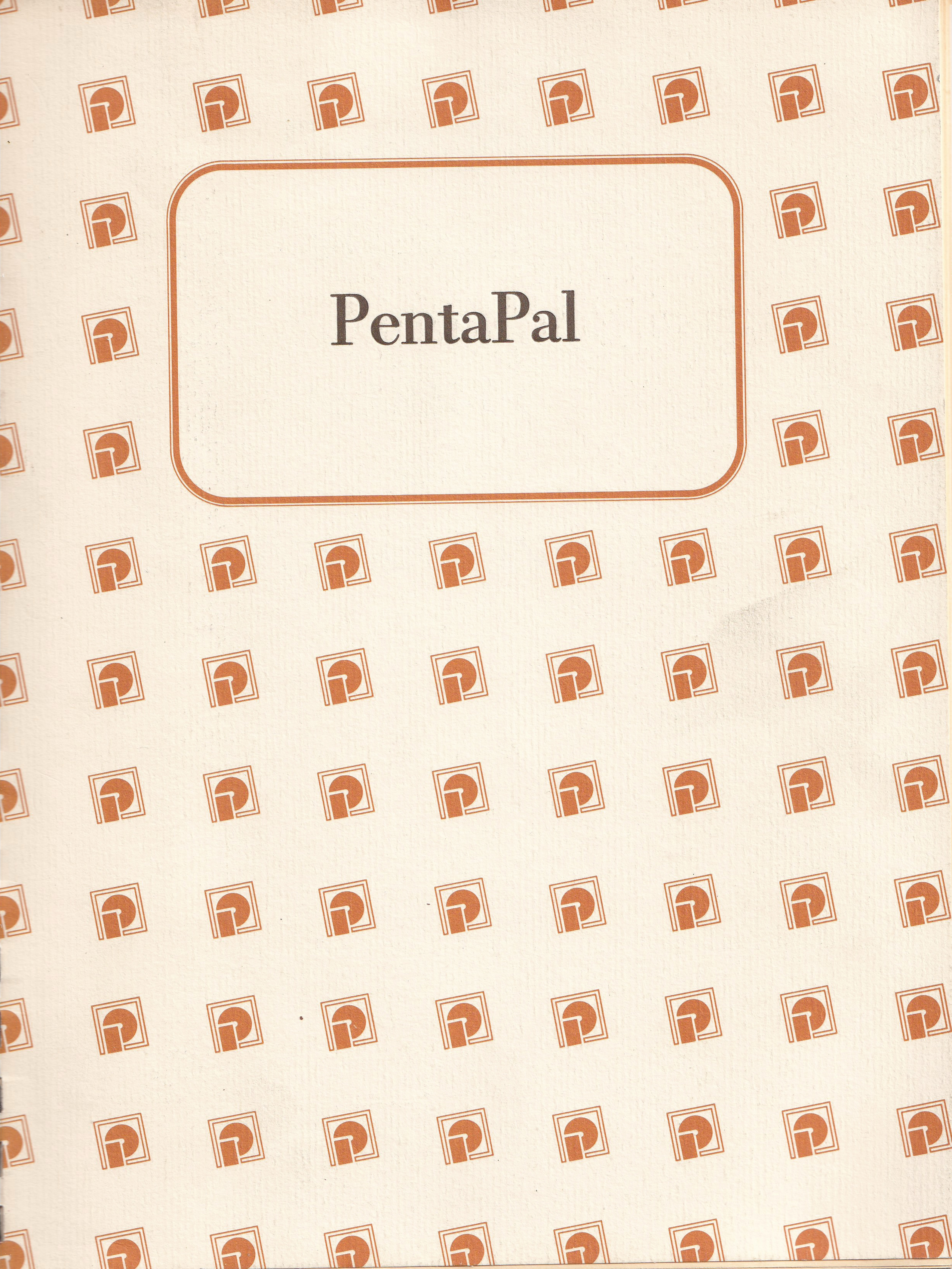

PentaPal

PentaPal

MANUAL

FOR

MARK-UP

AND

KEYBOARD INPUT

Penta Systems International, Inc.
1511 Guilford Avenue, Baltimore, Maryland 21202

This Manual represents the current state of the PentaPal Manual for Mark-Up and Keyboard Input. The utmost of care has been taken to make this Manual as accurate as possible. However, as with any Manual of this complexity, errors are possible.

We reserve the right to modify, delete, and/or add commands and/or capabilities at our discretion. We are not responsible for errors resulting directly or indirectly, from the use or mis-use of this Manual.

TABLE OF CONTENTS

Introduction

Glossary . . . . . Introduction 1
Glossary—Continued . . . . . Introduction 2
Markup . . . . . Introduction 3
Search and Replace List . . . . . Introduction 4
Search and Replace List Sample . . . . . Introduction 5

Section 1—General Commands

Define Operator 1–1
Mixed Font Specification 1–2
Family Font Specification 1–3
Combination Command 1–4
Combination Commmand Flexibility 1–6
End of Paragraph 1–7
Insert Name 1–8
Insert Date 1–9
Quad Left, Quad Right, Quad Center 1–10
Force Justify 1–11
Change Face 1–12
Change Point Size 1–13
Change Lead 1–14
Extra Lead/Reverse Lead 1–15
One Line Leading 1–16
End of Tape 1–17

Section 2—Miscellaneous Commands

Save Parameters/Restore Parameters 2–1
Cancel Flash/Reestablish Flash 2–2
Change Base 2–3
Change Set Size 2–4
Horizontal Movement 2–5
Zero Set . . . . . ***
Vertical Justification 2–6
Upper Rail/Lower Rail 2–7
Begin Oblique/Cancel Oblique 2–8
Secondary Mode/Normal Mode 2–9
Initialize Counter/Output Counter 2–10

Section 3—Indent Commands

Indent Hang 3–1
Indent Tape 3–3
Indent First Line 3–4
One Line Indent/Override 3–5
Indent Left (Unknown Amount) 3–6
Indent Right (Unknown Amount) 3–7
Wrap Left (Contouring) 3–8
Wrap Right (Countouring) 3–10

*** These Commands will be forth coming.

Section 4—Hyphenation/Justification Commands

Ragged Center 4–1
Ragged Left 4–2
Ragged Right 4–3
Conditional Hyphenation 4–4
Allow Hyphenation/Cancel Hyphenation 4–5
Change Justifiers 4–6
Change White Space 4–7
Band Alert 4–8
Allow Letterspace . . . . . ***
Increase/Decrease Character Width 4–9
Track Command 4–10
Over-Ride Track 4–12
Allow Kerning/Cancel Kerning 4–13

Section 5—Leader Commands

White Space Fill 5–1
Leader Fill 5–2
Variable Leader (for one line only) 5–3

Section 6—Universal Format Commands

Format Introduction 6–1
Define Universal Format/End Universal Format 6–2
Call Universal Format 6–3
Branch to Universal Format 6–4
Copy Merge 6–5
Format at Leading 6–6

*** These Commands will be forth coming.

Section 7—Tabular Commands

Define Tabular Column 7–1
Horizontal Tabular Setting 7–2
Vertical Alignment 7–4
Define Straddle Head 7–6
Establish Underscore (Linotron 202, APS, 8600 only) 7–8
Start/Cancel Underscoring 7–9
Rule Command (Linotron 202, APS, 8600 only) 7–10

Section 8—Variable Commands

Variable Introduction 8–1
Store Variables 8–2
Variable Addition 8–3
Variable Add W/Zero 8–4
Variable Subtraction 8–5
Variable Subtract W/Zero 8–6
Variable Times (Multiplication) 8–7
Variable Times W/Zero 8–8
Variable Division 8–9
Variable Division W/Zero 8–10
Variable Greater Than 8–11
Variable Greater W/Zero 8–12
Variable Less Than 8–13
Variable Less Than W/Zero 8–14

GLOSSARY

[= Begin command code is used to initiate all commands.

] = The end command code is used to terminate the specified field commands.

☐ = Em space.

☑ = En space—Ens are not necessarily one-half the width of ems.

⊗ = Thin space.

ragged = Ragged commands **DO NOT** automatically cancel hyphenation.

cancel ragged = Cancels all ragged commands.

define hyphenation = Defines hyphenation.

cancel hyphenation = Cancels hyphenation.

alphanumeric = Combination of letters and/or numbers only—no special symbols, i.e., &, ?, etc.

integer field = A command field consisting of numbers (no decimal point).

file = Usually used to mean a tape in disk storage.

tape = Normally means a division of copy or manuscript in one of the following forms (files):

1. punched into 6 level paper tape.
2. punched into 8 level paper tape.
3. stored on magnetic cassettes.
4. stored on magnetic tape reels.
5. stored on magnetic disk.

tape suffix = A one or two position alphanumeric field following the tape identification used to identify different versions of the same tape.

single key command = A one-keystroke command which is interpreted by **PentaPal** as a substitute for a combination of keystrokes. Most paper tape keyboards have a key called a "quad center key [QC." This single keystroke replaces the three keystrokes: [qc. Use of our **Search and Replace List** program permits the usage of any code-producing key on your keyboards as a single-key command.

⟨ ⟩ = Indicates a single key command is embedded.

⟨QL⟩ = A single key assigned to represent a quad left command.

⟨QR⟩ = A single key assigned to represent a quad right command.

⟨QC⟩ = A single key assigned to represent a quad center command.

⟨UR⟩ = A single key assigned to represent an upper rail command.

⟨LR⟩ = A single key assigned to represent a lower rail command.

⟨NT⟩ = A single key assigned to represent a next tab command.

⟨CM⟩ = A single key assigned to represent a copy merge command.

GLOSSARY—Continued

copy block terminator = The following commands are used to force a line ending and function as an end of paragraph command:
[ep
[qc ⟨QC⟩
[qr ⟨QR⟩

immediate line ending = The following will not affect the tape but will cause a line ending
[ql ⟨QL⟩
[fj

paragraph = A paragraph is considered to be any amount of copy followed by a [ep, [qc (⟨QC⟩), or [qr (⟨QR⟩).

paragraph status = A paragraph can have the following status (under the effect of a previous command):
All lines indented—[it___,___,]
All lines indented but not the first—[ih___,___]
All count lines indented—[il___,___] [ir___,___]
All lines indented—[ip___,___]
First line indented—[if___,___,___]
Lines indented following a—[li

justifier = Expandable word space (spaceband).

command field = The area reserved within a command to indicate where the user is to insert information necessary for the execution of the command (such as measure, point size, leading, number of lines, etc.).

quad line = A quad line is a blank line the depth of which will be whatever leading is in effect at the time.

system standards = The 120 items that are defined in your system standards data as constant values. They are generated during PentaPrep and can be overridden by some tape commands.

MARKUP

Mark-Up is an art and a science. Our large assortment of commands and the flexibility of their usage allows the markup person to be highly creative within the confines of the design of **PentaPal**.

Quality markup requires:

Anticipation—The ability (skimming through copy) to recognize areas which are unlike the *normal*. These areas will require special Mark-Up attention.

Concentration—Will ensure accurate Mark-Up and code selection and the right combination of both.

Imagination—You have a variety of commands available. Different combinations will frequently do the job.

Experience—This will broaden your base.

Remember, the more sophisticated the Mark-Up, the easier to keyboard.

The marriage of our software and your Mark-Up skills produce formats which reduces keyboarding to the simplest element possible.

Competent Mark-Up personnel have attained the ability to program our software which in turn programs for ouput to your phototypesetting machine.

SEARCH AND REPLACE LIST

SEARCH AND REPLACE LIST is designed to be used by Mark-Up personnel, keyboard personnel, computer personnel, and editing personnel. It is intentionally designed to be implemented at the computer to reduce the possibility of error. (See the PentaVue Manual for our 'on-screen' Search and Replace.

We call this feature "Search and Replace." On other systems it may be called "global search." Here are three uses of **SEARCH AND REPLACE LIST**.

- Assume you prepared material with all paragraphs indented one em space and it is changed by the customer to two em spaces. You simply instruct **PentaPal** to find all em spaces and convert them to two em spaces.
- A certain phrase is used throughout a job. Your keyboardist can assign any unused key on the keyboard to represent this phrase and thus reduce several words to one keystroke.
- Copy has been color-coded as follows:
  All major heads are color-coded blue and assigned format 1
  All secondary heads are color-coded red and assigned format 2
  All side heads are color-coded yellow and assigned format 3

In keyboarding you can put [jb for blue underscore and instruct **SEARCH AND REPLACE LIST** to recode it to the desired Universal format call ([u1]) which will insert the proper Universal format. In this example you have only saved one keystroke per code. The significant factor is that you have eliminated or reduced Mark-Up of the manuscript!

You can also instruct the system to change input material after keyboarding. Thus a phrase that repeats throughout a job can be changed via the computer without manually or visually editing the input tapes.

SEARCH AND REPLACE LIST makes every key on the keyboard programmable.

This feature has proven **SEARCH AND REPLACE LIST** to be a very valuable program. In addition to substantially reducing errors and keystrokes, it is very effective in salvaging input that was improperly coded.

Sample

Existing Codes	Replacement Codes
[ub	[u1]
[ur	[u2]
[uy	[u3]

Given a list similar to the above, the computer operator may search your tape for the existing codes and substitute the designated replacement codes.

SEARCH AND REPLACE LIST — SAMPLES

Operator Error

Throughout your copy you had the codes U.S. and an operator keyed U.S. but it was to be spelled-out.

Search for	Replace with
U.S.	United States

This Search and Replace will convert all (U.S.) to the correct codes.

Poor Manuscript

Throughout your copy you had the word lockwasher shown as *lock-washer, lockwasher*, and *lock washer*. When the job came back, the customer specified *"lockwasher"* as one word. You do not have to edit these changes. Instruct your computer operator to perform the following Search and Replace on this job.

Search for	**Replace with**
lock washer	lockwasher
Lock washer	Lockwasher
lock-washer	lockwasher
Lock-washer	Lockwasher

Time and Error Saver

The entire tape is a series of sentences separated by ellipsis. An ellipsis consists of a space, a period, a thin space, a period, a thin space, a period, and a space (end of sentence . . . Start next sentence). To save keystrokes and reduce error you decide to use a single key quad right command (⟨QR⟩) to represent the ellipsis. Instruct your computer operator to perform the following search and replace list:

Search for	**Replace with**
⟨QR⟩	.thin.thin.

Command Name: **Define Operator**

Input Code:	**Cancel Code:**
[do___,___,___,___]	None
[do___,___,___]	[te
[do___]	

Description:

The **first command field** is the Operator Number who punched the tape. The maximum number allowed is four digits.
The **second command field** is the Job Number you assigned the Job.
The **third command field** is the the Tape I.D.
The **fourth command field** is the Zip Code you assigned the Job if you are using Zip Codes.

Purpose:

To indicate which operator keyed the Tape, the Job Number, the Tape I.D. and the Zip Code. This command can be truncated at any point, **EXCEPT** after the second field. If fields two and three are filled in, the Job and Tape I.D. in this command will override the Job and Tape I.D. from the process request **During the Read Phase ONLY.**

Example:

Sample Input:

```
[do14,5001,1,5]
```

Sample Output: (on 20 picas)

Not Applicable.

Caution:

1. When used, this command **MUST** be the first entry in the tape and MUST precede any Universal Format calls.
2. This command **CANNOT** be used within a Universal Format.

Command Name: Mixed Font Specification

Input Code:
[mx__,__,__, . . . ,__]

Cancel Code:
[te

Purpose:
To specify *in loading sequence* (*position*) the fonts necessary to set a particular job. (Font information is loaded into **PentaPal** and assigned font numbers.)

Example:

Sample Input:

```
Assume there are twenty fonts loaded into [cf3]PentaPal[cfl], and that you need ten of
them to set a job. The ones you need are [cf3]PentaPal[cfl] fonts 16, 17, 18, 119, 1, 2,
22, 3, 7, and 8. The mixed font specification statement would look like this at the
very beginning of the tape:[ep [mx16,17,18,119,1,2,22,3,7,8]. If you wanted to
change to font 22 during keyboarding, you would key [cf7], because it occupies the
seventh position in this command. The seventh position is Helvetica Medium and so
that is why you see Helvetica Medium after the change face call.[ep
```

Sample Output: (on 20 picas)

Assume there are twenty fonts loaded into **PentaPal**, and that you need ten of them to set a job. The ones you need are **PentaPal** fonts 16, 17, 18, 119, 1, 2, 22, 3, 7, and 8. The mixed font specification statement would look like this at the very beginning of the tape: [mx16,17,18,119,1,2,22,3,7,8]. If you wanted to change to font 22 during keyboarding, you would key , **because it occupies the seventh position in this command. The seventh position is Helvetica Medium and so that is why you see Helvetica Medium after the change face call.**

Computer Use:
The computer uses the actual position of the fonts in this command to directly relate to the change face commands used when keying text. The identifying font number is assigned when the font widths are loaded into **PentaPal**.

Caution:
1. This command *must* precede all commands except the [do__,__,__,__] command.
2. The mix command may be used in a Universal Format and called as the first entry in the tape.
3. Failure to place this command in the proper place will result in not getting a visual in the tape. The visual is needed because PentaPi may add font numbers for automatic insertion of pi characters, etc.
4. This must be followed by a valid combination command and an end paragraph command.
5. Only one mix command per tape and it must be stated at the very beginning of the tape.

Command Name: Family Font Specifications

Input Code:
[fy__,__,__,__]

Cancel Code:
another [fy__,__,__,__] command
[te

Description:

The **first command field** is the *family of fonts* you wish to use, which consists of a maximum of 18 fonts. The **second command field** is the *face* number of the subsequent type, and corresponds to the *actual* positions within the family command.
The **third command field** is the *point size.*
The **fourth command field** is base *leading.*

Purpose:

To specify a family of fonts. One family consists of a group of fonts. Family font arrangements are applicable to CRT phototypesetters — Linotron 202, CG 8600 and APS MICRO 5. (Font information is loaded into **PentaPal** and assigned font numbers. Also groups of fonts are then assigned a family number.)

Example:

Sample Input:

```
Assume Times Roman family is assigned to family number 29. The fonts in the family
are Times Roman, Times Roman Italic, Times Roman Bold, Times Roman Bold Italic,
Times Roman, Times Roman Small Caps, etc. and in that position order. To use family
29 the keyboarding would look like this [fy29] and if you wished to use Times Roman
Bold [cf3] to get the third position of Times Roman Family 29.[ep
```

Sample Output: (on 20 picas)

Assume Times Roman family is assigned to family number 29. The fonts in the family are Times Roman, Times Roman Italic, Times Roman Bold, Times Roman Bold Italic, Times Roman, Times Roman Small Caps, etc. and in that position order. To use family 29 the keyboarding would look like this [fy29] and if you wished to use Times Roman Bold **to get the third position of Times Roman Family 29.**

Computer Use:

The computer uses the actual position of the fonts within the family in this command to directly relate to the change face commands used when keying text. The identifying font number is assigned when the font widths are loaded into **PentaPal**.

Caution:

1. This command **MUST** precede all commands except the [do__]. Unlike the Mixed Font Specifications command, the Family Font Specifications command [fy__,__,__,__] can be called at anytime within a tape. MX and FY cannot be used together.
2. A change face command **MUST** always follow a family command. The change face may be part of the family command, part of a [cc command, or simply a [cf command.

Command Name: **Combination Command**

Input Code:
[cc___,___,___,___]
(See description)

Cancel Code:
[cc___,___,___,___]
[cf___,___,___]
[cp___,___]
[cl___]
[te

Description:

The **first command field** is the *column measure* and is specified in picas and points. If no points are necessary, a comma follows the numbers representing picas. If points are used, a period or "p" must follow the numbers representing picas, then the numbers representing points.
The **second command field** is the *face* number of the subsequent type, and corresponds to the *actual* positions within the mixed font specification or family command.
The **third command field** is the *point size.*
The **fourth command field** is base *leading.*

Purpose:

This command is used to specify measure, face, point size, and leading to begin a tape and to allow changes within a tape.

Example:

Sample Input:

```
[cc20,1,8,10]We have now changed the measure to 20 picas, font position 1, 8 point
face on a 10 point body (leading).[ep[cc22,2,9]We have changed the measure to 22
picas, font position 2, and 9 point. We did not state the leading (fourth field) so
the leading of 10 point stays in effect.[ep[cc30,3]We now have changed the measure to
30 picas and the font position to 3. We did not change the point size (third field) or
the leading (fourth field) so the last stated point size and leading remain the
same.[ep[cc16]Finally we are changing only the measure to 16 picas. As you have
gathered by now you can state all 4 fields or the first 3 fields, or the first 2
fields or just the first field.[ep
```

Sample Output: (on 20 picas)

We have now changed the measure to 20 picas, font position 1,
8 point face on a 10 point body (leading).

*We have changed the measure to 22 picas, font position 2, and 9 point.
We did not state the leading (fourth field) so the leading of 10 point stays
in effect.*

**We now have changed the measure to 30 picas and the font position to 3. We did not
change the point size (third field) or the leading (fourth field) so the last stated point
size and leading remain the same.**

**Finally we are changing only the measure to
16 picas. As you have gathered by now you can
state all 4 fields or the first 3 fields, or the first
2 fields or just the first field.**

Command Name: **Combination Command—Continued**

Computer Use:

The computer uses the data in the command fields to replace the previously stated fields. The new fields will remain in effect until the end of the tape or until changed by one of the cancel codes shown above.

Caution:

1. This command **MUST** be followed by an end paragraph command ([ep) the first time it is used in a tape.
2. This command cannot be embedded within a line or paragraph, i.e., it must follow a copy block terminator or immediate line ending.
3. You cannot skip fields in [cc command starting from left side.

Combination Command Flexibility

(Rather than having to specify an entire 4-position field or having to key every pica measure, font position, point size, or leading as separate commands, you can combine them in any of the various ways illustrated below.)

Combination Command [cc__,__,__,__] (measure, face, pt. size, leading)	[cc__,__,__,__] Combination Command (measure, face, pt. size, leading)
Change Face [cf__,__,__] (face, pt. size, leading)	[cc__,__,__] Combination Command (measure, face, pt. size)
Change Face [cf__,__] (face, pt. size)	[cc__,__] Combination Command (measure, face)
Change Face [cf__] (face)	[cc__] Combination Command (measure)
Change Point [cp__,__] (pt. size, leading)	
Change Point [cp__] (pt. size)	
Change Leading [cl__] (leading)	
→	←
shorten from left field (change command and reduce one field)	**shorten from right field** (reduce one field do not skip through any field)

Command Name: End of Paragraph

Input Code:	**Cancel Code:**
[ep	None [te

Purpose:
This ends the paragraph by setting the last line quad left (or center, or right if setting ragged) and cancels or resets all codes applicable to the preceding paragraph. If using leaderfill or white space fill it will justify the line. (See caution below)

Example:

Sample Input:

```
this is the end of a paragraph.[epThis begins new paragraph.[ep
```

Sample Output: (on 20 picas)

this is the end of a paragraph.
This begins new paragraph.

Computer Use:
Tells the computer to quad the last line left if setting justified or ragged right. If setting ragged left, last line will quad right. If setting ragged center, last line will quad center.

Caution:

1. The end of paragraph terminator automatically cancels the following indention commands:
   Hanging Indention for Just One Paragraph ([oh___])
   Indent One Paragraph ([op___])
   Indent Left Unknown Amount ([li)
   Indent Right Unknown Amount ([ri)
   Also, reactivates commands on the first line of a paragraph for the following:
   Hanging Indention ([ih___])
   Indent First Line ([if___])
2. A [ep (no copy preceding it) will result in the output of a blank line of the depth corresponding to current leading in effect at that time.

Command Name:

Insert Name

Input Code:	**Cancel Code:**
[in	None [te

Purpose:

To place the tape identification assigned during read into any location in the tape—usually in a guide slug. It can also be set at end of each tape, just before the [te.

Example:

Sample Input:

```
This is a normal tape.[ep[in[ep This is the next paragraph of the tape and this is how
the [in command [in appears inside of a paragraph.[ep
```

Sample Output: (on 20 picas)

This is a normal tape.
KEYPAL$$$1
This is the next paragraph of the tape and this is how the [in
command KEYPAL$$$1 appears inside of a paragraph.

Computer Use:

The computer remembers the job and tape name and at each appearance of a [in command, inserts the full 10 position tape identification.

Caution:

1. If the command is keyed in the secondary mode it will be output in that position, possibly making it unreadable.
2. When using [in in the middle of a sentence or at the beginning of a line, it is set as if it were a word—It DOES NOT have any quadding command attached to it as an automatic function.

Command Name: **Insert Date**

Input Code:	**Cancel Code:**
[id	None [te

Purpose:

To place the current date of processing (during the JUSTIFY phase) and time into any location in the tape—usually at the beginning of the tape immediately following the first family, full [cc, an [ep command, or in a guide slug. It can also be set at end of each tape, just before the [te.

Example:

Sample Input:

```
This is a normal tape.[ep[id[ep This is the next paragraph of the tape and this is how
the [id command [id appears inside of a paragraph.[ep
```

Sample Output: (on 20 picas)

```
This is a normal tape.
05-28-81 17-48-33
This is the next paragraph of the tape and this is how the [id
command 05-28-81 17-48-33 appears inside of a paragraph.
```

Computer Use:

The computer remembers the date and time of processing (during the JUSTIFY phase) according to the real time clock and at each appearance of a [id command, inserts the date and time.

Caution:

1. If the command is keyed in the secondary mode it will be output in that position, possibly making it unreadable.
2. When using [id in the middle of a sentence or at the beginning of a line, it is set as if it were a word—It DOES NOT have any quadding command attached to it as an automatic function.

Command Name:

Quad Left, Quad Right, Quad Center

Input Code:	**Cancel Code:**
[ql Quad Left	None
[qr Quad Right	None
[qc Quad Center	None
	[te

Purpose:

Quad commands are used to place a line of copy in the appropriate position on the line of the prestated measure. You may use the single key functions on the PentaVue Terminals.

Example:

Sample Input:

```
This is a line to quad left using the End paragraph key(¶)[epThis is a line to quad
left using the Quad Left key[qlThis is a line to quad right using the Quad Right
key[qrThis is a line to quad center using the Quad Center key[qc
```

Sample Output: (on 20 picas)

```
This is a line to quad left using the End paragraph key(¶)
This is a line to quad left using the Quad Left key
                    This is a line to quad right using the Quad Right key
        This is a line to quad center using the Quad Center key
```

Computer Use:

The computer directs all characters preceding the quad commands in the appropriate direction on output.

Caution:

1. Quad commands will not cancel hangs or destroy paragraph status. (See Glossary Page Introduction 2, paragraph and paragraph status.
2. Quad left, in particular, can be used instead of [ep to allow paragraph indents to continue through several paragraphs. (See hanging indents)
3. If copy for headings exceeds one justified line, it is suggested that you precede the information with ragged commands and terminate paragraphs with [ep.

Command Name: **Force Justify**

Input Code:
[fj

Cancel Code:
None
[te

Purpose:

Indicates that the line just typed is to be justified, beginning and ending flush with the left and right margins of the measure in effect.

Example:

Sample Input:

```
This is a typical paragraph of text that you are setting. Note that normal
justification is taking place on the line before the [fj command. Now you decide to
justify right here[fjor anywhere in the text. Notice that after the [fj normal
justification also takes place.[ep
```

Sample Output: (on 20 picas)

This is a typical paragraph of text that you are setting. Note that normal justification is taking place on the line before the [fj command. Now you decide to justify right here
or anywhere in the text. Notice that after the [fj normal justification also takes place.

Computer Use:

The computer will take the line up to the [fj and distribute equally enough space between all words to justify the line.

Caution:

1. Notice that sample is letterspaced the maximum amount allowed by our system standards. If letterspacing is not desired for this line but is desired for remaining lines, precede line by cancel letterspace command and follow it with allow letterspace command.
2. This command will force justify the line and may cause excess spacebands on loose lines.
3. [fj must be followed by [ep if there is an indent specification fit.
4. [fj must be followed by [ep *if* there is an indent specification requiring a [ep to cancel it.
5. On tight lines it forces the space bands to zero and over sets the measure.

Command Name: **Change Face**

Input Code:
[cf__,__,__]
(See Description)

Cancel Code:
another [cf command or a [cc command using the first two fields.
[te

Description:

The **first command field**, a one or two digit field from 1 to 18 (depending upon the limitation of the output machine), is the *actual position* of the desired font in the mixed font specification command (or family font specification command), not the font number.
The **second command field** is the *point size*.
The **third command field** is the *base leading*.

Example:

Sample Input:

```
[cc20,1,8,10]You are setting type in Times Roman at 20 pi, 8 point type on 10 pt
leading.[ep[cf2,10,12]You have changed font position to 2 which is Times Roman
Italic, 10 point type on 12 pt leading.[ep[cf3,12]You have changed font position to 3
which is Times Roman Bold, and changed the point size of the type to 12. The 12 point
leading stays in effect. [cf4]Now in the middle of this paragraph you change font
position to 4 which is Times Roman Bold Italic.[ep
```

Sample Output: (on 20 picas)

You are setting type in Times Roman at 20 pi; 8 point type on 10 pt leading.

You have changed font position to 2 which is Times Roman Italic, 10 point type on 12 pt leading.

You have changed font position to 3 which is Times Roman Bold, and changed the point size of the type to 12. The 12 point leading stays in effect. *Now in the middle of this paragraph you change font position to 4 which is Times Roman Bold Italic.*

Caution:
Rail commands can be used to change font position with one keystroke. (See rail command description.)

Command Name: **Change Point Size**

Input Code:	**Cancel Code:**
[cp—,—]	full [cc, full [cf, or another [cp command
(See Description)	[te

Description:

The **first command field** is the point size you desire for the following type.
The **second command field** (if present) is the desired base leading in increments of your phototypesetter.

Purpose:

To change point size in a tape and leading if desired.

Example:

Sample Input:

```
This is the end of a paragraph.[ep[cp6,8]This is a footnote that is to be set in 6
point type on 8 point leading.[ep[cp10,12.5]This paragraph is to be 10 point type on
12 and one half point leading.[ep[cp8]This paragraph is set in 8 point type but
remains on 12 and one half-point leading. [cp14]This paragraph is set in 14 point
type. Notice that no change in leading was issued, therefore the size of the type is
greater than the[epleading.[ep
```

Sample Output: (on 20 picas)

This is the end of a paragraph.

This is a footnote that is to be set in 6 point type on 8 point leading.

This paragraph is to be 10 point type on 12 and one half point leading.

This paragraph is set in 8 point type but remains on 12 and one half-point leading. This paragraph is set in 14 point type. Notice that no change in leading was issued, therefore the size of the type is greater than the leading.

Computer Use:

The computer uses the change point size command to adjust the character widths necessary to compute output lines during justification.

Caution:

1. If you change point size only, you do not affect the present leading.
2. A change point size will return your set size back to what is specified in your family. (CRT)

Command Name: **Change Lead**

Input Code:	**Cancel Code:**
[cl___]	full [cc, full [cf, full [cp or another [cl command [te

Purpose:

To change the base leading in a tape, in increments of your phototypesetter.

Example:

Sample Input:

```
You are now looking at a paragraph being set in 8 point type on a 10 point leading.
The next paragraph will change the leading once we reach this line.[ep[cl28] Assuming
that the type size is 8 point, this copy would be set 8 on 28 (28 points baseline to
baseline allowing 20 points of space between the lines).[ep
```

Sample Output: (on 20 picas)

You are now looking at a paragraph being set in 8 point type on a 10 point leading. The next paragraph will change the leading once we reach this line.

Assuming that the type size is 8 point, this copy would be set 8

on 28 (28 points baseline to baseline allowing 18 points of space

between the lines).

Caution:

1. This command does not effect the point size.
2. If the change lead command ([cl___]) is key in the middle of a line, it will **NOT** take effect until the next line.

Command Name: **Extra Lead/Reverse Lead**

Input Code:	**Cancel Code:**
[el___] Extra Lead	None
[el-___] Reverse Lead	[te
(See Description)	

Description:

This **command field** describes the number of points of leading. If a "p" is entered after the number you will get the Extra Lead/Reverse Lead in picas and points. If no "p" is stated it defaults to points.

Purpose:

This command can be keyed anywhere in text. The extra lead and lead reverse takes effect at the point it falls during justification. The leading within the command is the desired amount of space in picas and/or points and can be one thru four digits with a maximum of: Linotron 202 = 8,191 points, VIP = 16,383 points, CG 8600 = 4,095. If quarter point leading is desired use .25 or .3 . If half point leading is desired, use .50 or .5 . If three-quarter point leading is desired, use .75 or .8 .

Example:

Sample Input:

```
Chapter one[qc[el12.5]Introduction[qc[el10][cfl]Part 1[ql[el3p]The M-1 is a gas
operated, air cooled, clip fed, semi-automatic shoulder weapon, weighing 9.5
pounds.[ep
```

Sample Output: (on 20 picas)

Chapter one

Introduction

Part 1

The M-1 is a gas operated, air cooled, clip fed, semi-automatic
shoulder weapon, weighing 9.5 pounds.

Computer Use:

The computer inserts an extra lead/reverse lead command to the maximum of your phototypesetter.

Caution:

1. The exception to this is that extra lead/reverse lead specified in TABULAR will be effected at the beginning of the line in that tabular column.
2. The maximum limit and leading increment differs for certain CRT's.

Command Name: **One Line Leading**

Input Code:
[ol__]

Cancel Code:
None
[te

Purpose:
This command is used whenever you wish to change leading for one line only. **Takes effect for the line in which it is called.**

Example:

Sample Input:

```
Set this paragraph with a base leading of 10 points (8 on 10), but change the leading
to 18 points for one line. When this command is encountered the entire justified
line receives a leading change. This occurs right here [ol18]even if it is in the
middle of a justified line instead of the point at which the justified line
begins.[ep
```

Sample Output: (on 20 picas)

Set this paragraph with a base leading of 10 points (8 on 10), but change the leading to 18 points for one line. When this command is encountered the entire justified line receives a leading change.

This occurs right here even if it is in the middle of a justified line instead of the point at which the justified line begins.

Computer Use:
The computer will output the leading change where indicated for one line only.

Caution:
1. Does not change leading permanently.
2. Will not change the point size automatically. Use [op with [ol if both require changing.

Command Name: **End of Tape**

Input Code:	**Cancel Code:**
[te	None

Purpose:
To indicate the end of the input tape. Cancels all codes in effect.

Computer Use:
Stores all statistics for the operator indicated when the tape was loaded.

Caution:

1. If this command is absent from the input tape, when the tape is being read, PentaPal will insist the computer operator to enter the [te.
2. An end tape command placed in the middle of a tape will cause everything from that point on to be deleted if a "Renumber" request was asked for in the process stage.

Command Name: Save Parameters / Restore Parameters

Input Code:	**Cancel Code:**
[sp Save Parameters	[rp Restore Parameters [te

Purpose:

To be able to save existing (or last stated) family, measure, face, point size, leading, set size, slant, base line, tracks, increase widths, and band values (minimum, optimum, maximum before hyphenation, kerning, letterspacing) currently in effect.

Example:

Sample Input:

```
This is last line of current status which is being set to a column measure of 20
picas.[ep[sp[cc16,4,10,12]This is now 16 picas and we will change back to existing
status with the restore parameters command.[ep[rpThis will revert back to existing
status automatically to 20 picas due to the restore parameters command.[ep
```

Sample Output: (on 20 picas)

This is last line of current status which is being set to a column measure of 20 picas.

This is now 16 picas and we will change back to existing status with the restore parameters command.

This will revert back to existing status automatically to 20 picas due to the restore parameters command.

Caution:

1. Any changes to these fields after the Save Parameter Command ([sp) will be overridden when you use the Restore Parameters Command ([rp).
2. This command **does not** hold any hangs, indents, etc.
3. **Do not** use this command while in tabular.

Command Name:

Cancel Flash/Reestablish Flash

Input Code:	**Cancel Code:**
[xf Cancel Flash	[rf Reestablish Flash
	[te

Purpose:

To allow space for a character or series of characters but not actually flash those characters on the phototypesetter. Flash on/off capabilities not required on phototypesetters.

Example:

Sample Input:

```
Allow enough room for a cap [xfT[rf but do not flash as we will insert special art
work.[ep
```

Sample Output: (on 20 picas)

Allow enough room for a cap but do not flash as we will insert
special art work.

Caution:

1. A missing [rf will cause flash off to be in effect for the rest of the tape or until another [rf is found.
2. This command should not be used in tabular. Penta does not use reverse leading while in the tabular mode so it could cause blank columns.

Command Name:

Change Base

Input Code:	**Cancel Code:**
[cb__] Below Base Line	[cb0]
[cb-__] Above Base Line	[te

Description:

A **numeric field** *in points* indicating how far to *move down* from the baseline or if preceded by a hyphen to *move up* from the baseline.

Purpose:

To reverse or advance the base line setting in a line by using the immediate film movement capabilities of your phototypesetter.

Example:

Sample Input:

```
Price per unit [cb-6]$[cb0][cp14]9.[cb6][cp8]99[cb0] each.[ep
```

Sample Output: (on 20 picas)

Price per unit $^{\$}9._{99}$ each.

Computer Use:

The computer will issue the correct commands to utilize your phototypesetter's reverse lead capability.

Caution:

1. Your phototypesetter must have reverse lead capabilities to use this command.
2. This command **is not** accumulative.
3. The change base command and the baseline deflection defined in the family are accumulative.

Command Name: **Change Set Size**

Input Code:	**Cancel Code:**
[cs__]	another [cs__], [cp__,__] or a new full [fy command [te

Purpose:

To change the set width of the type but not the height. The command has 1 field. This is the point size whose width you want to use.

Sample Input:

```
We are setting a 10 point type.[ep[cs14]We are still setting in type that is 10 points
high but has the width of 14 point type.[ep
```

Sample Output: (on 20 picas)

We are setting a 10 point type.
We are still setting in type that is 10
points high but has the width of 14
point type.

> **Caution:**
> Can only be used on photo machines that have the ability to alter the characters.

Command Name: **Horizontal Movement**

Input Code:	**Cancel Code:**
[hm___] Move from Left	Another [hm___]
[hm-___] Move from Right	[te

Purpose:

To do a horizontal movement either left or right. The command field is expressed in picas and points. A [hm1p] would cause a horizontal move of 1 pica to the right and a [hm-2p] would cause a horizontal move of 2 picas to the left.

Sample Input:

```
If you wish to do a horizontal move of 3 picas to the right at this point [hm3p] you
would have keyed the command right where you desired it.[epIf you wish to do a
horizontal move of 3 picas to the left at this point you would have keyed the command
right [hm-3] where you desired it.[ep
```

Sample Output: (on 20 picas)

If you wish to do a horizontal move of 3 picas to the right at this
point   you would have keyed the command right where
you desired it.

If you wish to do a horizontal move of 3 picas to the left at this
point you would have keyed the commawhderighbu desired it.

Caution:

1. The amount of the horizontal move will not be displayed on the PentaVue Terminal.
2. Only available if your phototypesetter has this capability.

Command Name: **Vertical Justification**

Input Code:	**Cancel Code:**
[vm___] Vertical Measure	[vj Vertical Justify
[vb Vertical Band	[te

Purpose:

To allow Vertical Justification of a block of copy to a certain depth. It counts all leading stated between the [vm command and the [vj command.

Example:

Sample Input:

```
[vm8p]Centered Head[qc[vbText starts here . . . .[ep [vb[vbThis is the last line of
the text copy and should be on baseline of the 8 pica depth.[ep[vj
```

Sample Output: (on 20 picas)

Centered Head

Text starts here . . . .

This is the last line of the text copy and should be on baseline of
the 8 pica depth.

Computer Use:

The computer in this example will make an 8 pica deep area by using variable length vertical bands. Since two of them are together it will calculate and generate double the space generated by a single [vb.

Caution:

1. If the type exceeds the measure, the area will be set to whatever depth it will reach without vertical bands.
2. You must have a [vj for every [vm or command will not work properly.
3. Must be stated in picas and points only.
4. The maximum depth is the same as Extra Lead/Lead Reverse.

Command Name: **Upper Rail/Lower Rail**

Input Code:	**Cancel Code:**
< Upper Rail	> Lower Rail [cf__] [te

Purpose:

To change from a odd position to an even position, within a family or the mix command.

Sample Input:

```
You are keying out of position nine in the family or mix command and you key an upper
rail here <with just a single key we have changed to the tenth position until a lower
rail is keyed here> now back to position nine.[ep
```

Sample Output: (on 20 picas)

You are keying out of position nine in the family or mix command
and you key an upper rail here *with just a single key we have changed
to the tenth position until a lower rail is keyed here* now back to position
nine.

Computer Use:

The upper rail command will permit an operator to change from an odd position to an even position. The lower rail goes from the even position back to the odd. If you are in an even position and key an upper rail it will be ignored, if you are in an odd position and key a lower rail it will be ignored.

Command Name: **Begin Oblique/Cancel Oblique**

Input Code:	**Cancel Code:**
[bo Begin Oblique	[xo Cancel Oblique None [te

Purpose:

To set type with a slant. To create a pseudo italic.

Sample Input:

```
We are setting in a roman face and the oblique command is issued here. [boWe have not
changed face but now the type is slanted to here[xo where it is cancelled.[ep
```

Sample Output: (on 20 picas)

We are setting in a roman face and the oblique command is issued here. *We have not changed face but now the type is slanted to here* where it is cancelled.

Caution:

Your phototypesetter **MUST** have the ability to slant characters.

Command Name: Secondary Mode/Normal Mode

Input Code:	**Cancel Code:**
[sm Secondary Mode	[nm Normal Mode
	None
	[te

Purpose:
To set type from the secondary mode. The setting up of a Secondary Mode depends upon your typesetters. Therefore, because of the various ways in which this command may be used, it is not possible to output samples. (Normally, set-up for keying caps and small caps).

This is also explained in the PentaPrep Manual.

Command Name: **Initialize Counter/Output Counter**

Input Code:
[ic___,___,___] Initialize Counter
[oc Output Counter

Cancel Code:
another [ic
[te

Description:

The **first command field** is the specified number you wish to start with.
The **second command field** is the number of positions you wish to fill before the number.
The **third command field** is if they are to be fill with zeros or en's which are added to the left of the numbers. A zero (0) is for zeros and a one (1) is for en's.

Purpose:

To initialize an incremented counter to start at a specified number and then output that number in the tape whenever desired.

Sample Input:

```
[ic14,3,1][oc[oc[oc[oc[oc[oc[oc[ep
[ic998,4,1][oc[ep[oc[ep[oc[ep[oc[ep
[ic1,5,0][oc[ep[oc[ep[oc[ep[oc[ep[oc[ep[oc[ep
```

Sample Output: (on 20 picas)

```
 14 15 16 17 18 19 20
 998
 999
1000
1001
00001
00002
00003
00004
00005
00006
```

Computer Use:

The [ic command is used to initialize the counter to start with a specified number. The second field in the ic command is the number of digits to output. This can be from 1 to 5. If the number being output has less than the specified number of digits zeros or ens will be output to the left of the number. Each time the command oc is encountered the number is output and the counter incremented by 1. The maximum number that can be output is 32,767.

Command Name: **Indent Hang**

Input Code:
[ih___] left indent only
[ih0,___] right indent only
[ih___,___] left, right indents

Cancel Code:
[ih0], or Another [ih command
[te

Purpose:

To set all paragraphs flush left and right for first line and specified indents (in picas/pts. or ens/thins) for remaining lines of all paragraphs until canceled.

Example:

[ih2p6] = 2 picas and 6 points on left only
[ih0,3] = 3 (whatever your system standards table reads) on the right only
[ih2n1] = 2 ens and 1 thin on left only
[ih4p,2p] = 4 picas on the left and 2 picas on the right

Sample Input:

```
[ih6p]This paragraph is set with an indent hang on the left of 6 picas. The first line
will be flush and all subsequent lines indent on left 6 picas.[ep[ih3p,3p]This starts
a new paragraph where the first line will be flush but subsequent lines will indent 3
picas on the left and 3 picas on the right. Remember that picas are fixed measures
and ens vary with the point size of the font.[ep[ih0,4p6]This final paragraph will be
indented on the right only the amount of 4 picas and 6 points. The first line will be
flush. The last line of this paragraph will be terminated with a quad left to
demonstrate the effect.[qlAs you notice, this is the beginning of another and the
final paragraph in the sample input for indent hangs. The first line of this
paragraph is equal to the indent thus showing you that quad left holds a hang. I will
now cancel the indent hang at this time.[ep[ih0]
```

Sample Output: (on 20 picas)

```
This paragraph is set with an indent hang on the left of 6 picas.
                      The first line will be flush and all subsequent
                      lines indent on left 6 picas.
This starts a new paragraph where the first line will be flush but
           subsequent lines will indent 3 picas on the left
           and 3 picas on the right. Remember that picas
           are fixed measures and ens vary with the point
           size of the font.
This final paragraph will be indented on the right only the amount
of 4 picas and 6 points. The first line will be flush.
The last line of this paragraph will be terminated
with a quad left to demonstrate the effect.
As you notice, this is the beginning of another and
the final paragraph in the sample input for indent
hangs. The first line of this paragraph is equal to
the indent thus showing you that quad left holds
a hang. I will now cancel the indent hang at this
time.
```

Command Name: **Indent Hang—Continued**

Computer Use:

The computer uses the specified amount in each field to create the desired indents. Each field can be stated in picas, picas and points, points or ens, ens and thins, and just thins.

Caution:

If you wish to cancel either hang and keep the remaining one in effect, use [ih0,__p__] for cancel left and [ih__p__,0] for cancel right (filling in the blanks with current hang).

Command Name: **Indent Tape**

Input Code:
[it___,___] left, right indents
[it___] left indent only
[it0,___] right indent only

Cancel Code:
[it0],
another [it___,___]
[te

Purpose:

This command will indent all lines (in picas/pts. or ens/thins) until cancelled.

Example:

[it3p1,2p] = 3 picas and 1 point on the left and 2 picas on right
[it0,4p1] = 4 picas and 1 point on right only
[it4p1] = 4 picas and 1 point on the left only

Sample Input:

```
The matter now being typeset is not indented 2 picas on the left and 4 picas on the
right.[ep[it2p0,4p0]From here on the type will be indented completely (that is,
every line) on the right and left until a cancel command or another indent tape
command is encountered.[ep[itlp]Notice this paragraph is indented on the left l pica
and nothing on the right. At the end of this sentence there will be a cancel command
which will make the following paragraph flush to left and right margin.[ep[it0]This
line or paragraph, whatever it justifies to, will be flush left and right.[ep
```

Sample Output: (on 20 picas)

The matter now being typeset is not indented 2 picas on the left and 4 picas on the right.

From here on the type will be indented completely (that is, every line) on the right and left until a cancel command or another indent tape command is encountered.

Notice this paragraph is indented on the left 1 pica and nothing on the right. At the end of this sentence there will be a cancel command which will make the following paragraph flush to left and right margin.

This line or paragraph, whatever it justifies to, will be flush left and right.

Computer Use:

The computer puts this command into effect immediately and continues in effect until cancelled.

Caution:

If you wish to cancel either indent and keep the remaining one in effect, use [it0,___p___] for cancel left and [it___p___,0] for cancel right (filling in the blanks with the current indent).

Command Name: **Indent First Line**

Input Code:	**Cancel Code:**
[if__,__,__]	[if0] or another [if__,__,__]
(See Description)	[te

Description:

The **first command field** is the left indention desired on the first line of all following paragraphs.
The **second command field** is the right indention desired on the first line of all following paragraphs.
The **third command field** is the amount of leading desired between paragraphs.
The first and second command fields are specified in picas and points, or ens and thins, and the third data field is specified in points (of *additional* leading desired).

Purpose:

To indent the first line of every paragraph and output extra lead between paragraphs automatically.

Example:

[if1p,1p6,4] = 1 pica indent on left, 1 pica 6 points on right and 4 points white space between paragraphs
[if2n] = 2 en indent on left only
[if1p,0,6] = 1 pica indent on left, nothing on the right and 6 points between paragraphs

Sample Input:

```
[if2p,0,10]The next group of paragraphs will have the first line indented 2 picas
with 10 points of white space between each paragraph.[epThis paragraph will also
have the first line indented 2 picas with 10 points of white space between each
paragraph.[ep[if0]The above paragraphs were set using the Indent First Line Command.
Preceding this paragraph is a cancel command so the contents hereof are flush to the
left margin.[ep
```

Sample Output: (on 20 picas)

The next group of paragraphs will have the first line indented 2 picas with 10 points of white space between each paragraph.

This paragraph will also have the first line indented 2 picas with 10 points of white space between each paragraph.
The above paragraphs were set using the Indent First Line Command. Preceding this paragraph is a cancel command so the contents hereof are flush to the left margin.

Computer Use:

The computer uses the first and second parameters of this command to indent the first line of every paragraph, left and right respectively. The last entry is used to add extra leading between every paragraph. These parameters are stored for the entire tape unless changed or canceled.

Caution:

1. Does not recognize the first line after a [ql or [fj command as a new paragraph—consequently the next paragraph would be flush or indented the keyed amount.
2. This command **CANNOT** be used with [li or [ri commands.

Command Name: One Line Indent/Override

Input Code:	**Cancel Code:**
[oi__,__] left, right indents	After one use.
[oi__] left indent only	Another [oi__,__]
[oi0,__] right indent only	[te

Purpose:

To indent (in picas/pts or ens/thins) for one line only.
To override any existing indent in process (such as [ih, [oh, [it, [ip or [if) for one line only.

Example:

Sample Input:

```
[ihlp][it0,3p]This is a paragraph that is a description of a particular item set on 20
picas, 1 pica indent hang on left with 3 picas indent on the right as allowance for
the price which is to appear on the last line flush right from the copy. The price of
this item is[oilp,0][ws$9.39[epAs you can see, a white space fill command was used
after the one line indent command to achieve the position of the price to the far
right.[ep
```

Sample Output: (on 20 picas)

```
This is a paragraph that is a description of a particular
   item set on 20 picas, 1 pica indent hang on left with
   3 picas indent on the right as allowance for the price
   which is to appear on the last line flush right from
   the copy. The price of this item is                      $9.39
As you can see, a white space fill command was used
   after the one line indent command to achieve the
   position of the price to the far right.
```

Computer Use:

This command takes effect immediately for the line in which it appears.

Caution:

1. This command will cause an Error in JUSTIFY if it appears in a line and carries too great of an indent.
2. This command is to Override **ONLY** 1 (one) indent. If more than 1 (one) indent is in effect it will cause an Error in the JUSTIFY phase.

Command Name: Indent Left (Unknown Amount)

Input Code:	**Cancel Code:**
[li	[ep [te

Purpose:

This command is used to create a hanging indention at an arbitrary point in text from the left margin.

Example:

Sample Input:

```
End of paragraph.[ep[el12]Description: [liThe M-1 rifle is a gas operated, air
cooled, clip fed, semi-automatic, shoulder weapon weighing 9.5 pounds.[epParagraph
of text. This paragraph goes flush left.[ep
```

Sample Output: (on 20 picas)

```
End of paragraph.

Description: The M-1 rifle is a gas operated, air cooled, clip fed,
             semi-automatic, shoulder weapon weighing 9.5
             pounds.
Paragraph of text. This paragraph goes flush left.
```

Computer Use:

The computer calculates the indention up to the point where the command is encountered and hangs the remainder of lines until a [ep is found.

Caution:

1. Once another [li command is issued you cannot return to the former [li command.
2. Indent can be held by using [ql or [fj rather than a [ep at the end of a paragraph.

Command Name: **Indent Right (Unknown Amount)**

Input Code:	**Cancel Code:**
[ri	[ep [te

Purpose:

This command is used to create a hanging indention at an arbitrary point in text from the right margin.

Example:

Sample Input:

```
End of paragraph.[ep[el12]Description:□The M-1 rifle [ri is a gas operated, air
cooled, clip fed, semi-automatic, shoulder weapon weighing 9.5 pounds. [epParagraph
of text. This paragraph goes flush left.[ep
```

Sample Output: (on 20 picas)

End of paragraph.

Description: The M-1 rifle is a gas operated, air cooled, clip fed,
semi-automatic, shoulder
weapon weighing 9.5
pounds.
Paragraph of text. This paragraph goes flush left.

Computer Use:

The computer calculates the indention up to the point where the command is encountered and hangs the remainder of lines until a [ep is found.

Caution:

1. Once another [ri command is issued you can not return to the former [ri command.
2. Indent can be held by using [ql or [fj rather than a [ep at the end of a paragraph.

Command Name: Wrap Left (Contouring)

Input Code:	**Cancel Code:**
[wl___,___, . . . ___,___] (See description)	[wl0,0] Another [wl Satisfied depth [te

Description:

This command is stated in sets of instructions, each set defining down and across coordinates. A maximum of 25 sets can be stated at a time. Each coordinate value is separated by a comma.

The **first command field** of the set (down coordinate) is the depth, in picas and points, as measured from the top of the contour.

The **second command field** of the set (across coordinate) is the indent, in picas and points, as measured from the margin.

The coordinate values are ALWAYS measured from their relative starting points.

Purpose:

To define a series of down and across coordinates to be used in setting a contour. The shape of the contour reflects the coordinates set up in the command string. The contour between each coordinate point is depicted as an imaginary straight line connecting the two points. Thus, the final contour shape will consist of a series of straight lines connecting each coordinate.

In order to compensate for the possibility of ascenders and descenders occurring in the text and to take into consideration the shape of the contour, the system will automatically go through a 3-step calculation in order to maximize the indent to be used on each line. The indent is calculated by the following steps:

1. The indent, along the contour, is first calculated at a point of 75% of the point size above the base line (i.e., at the ascender);
2. The indent, along the contour, is then calculated at a point of 20% of the point size below the base line (i.e., at the descender);
3. The indent of any down and across coordinates that fall in the range between steps 1 and 2.

Example:

Sample Input:

```
[cc22,1,8,10][wl0,2p6,7,4p,12,5p10,16,3p,21,11p]This text is being set at a measure
of 22 picas. The point size is 8 pt. and the leading is 10 pt. A wrap left command is
issued at the beginning of the first paragraph defining 5 sets of down and across
coordinates.[epThe first set defines a 0-pica depth (down coordinate) to be indented
on the left and 2 picas and 6 points (across coordinate). The first line begins with
a proportionate amount of indent and gradually increases for every line until the
specified amount of 2 picas and 6 points is reached at a point to coincide with the
specified depth of 3 picas.[ep[cf2,12,14]The second set picks up where the first set
left off and defines another down coordinate of an additional 4-pica depth (7 picas
from the top) and an across coordinate of an additional 1 pica/6 points indent (4
picas from the margin). Again, this 4-pica depth will be skewed an appropriate
amount on each line until the 1 pica/6 point indent is exhausted to coincide with the
bottom of the 4-pica depth.[ep[cf1,10,12]The next set of coordinates defines an
additional 5-pica depth (12 picas from the top) and an additional 1 pica and 10
points indent (5 pica/10 points from the margin). After this 5-pica depth is
satisfied the remaining sets are executed after which the measure of 22 picas will
be resumed automatically.[ep
```

Command Name: **Wrap Left (Contouring)—Continued**

Sample Output: (on 20 picas)

This text is being set at a measure of 22 picas. The point size is 8 pt. and the leading is 10 pt. A wrap left command is issued at the beginning of the first paragraph defining 5 sets of down and across coordinates.

The first set defines a 0-pica depth (down coordinate) to be indented on the left 2 picas and 6 points (across coordinate). The first line begins with a proportionate amount of indent and gradually increases for every line until the specified amount of 2 picas and 6 points is reached at a point to coincide with the specified depth of 3 picas.

The second set picks up where the first set left off and defines another down coordinate of an additional 4-pica depth (7 picas from the top) and an across coordinate of an additional 1 pica/6 points indent (4 picas from the margin). Again, this 4-pica depth will be skewed an appropriate amount on each line until the 1 pica/6 points indent is exhausted to coincide with the bottom of the 4-pica depth.

The next set of coordinates defines an additional 5-pica depth (12 picas from the top) and an additional 1 pica and 10 points indent (5 pica/10 points from the margin). After this 5-pica depth is satisfied the remaining sets are executed after which the measure of 22 picas will be resumed automatically.

Command Name: Wrap Right (Contouring)

Input Code:	**Cancel Code:**
[wr__,__, . . . __,__]	[wr0,0]
(See description)	Another [wr
	Satisfied depth
	[te

Description:

This command is stated in sets of instructions, each set defining a down and across coordinate. A maximum of 25 sets can be stated at a time. Each coordinate value is separated by a comma.

The **first command field** of the set (down coordinate) is the depth, in picas and points, as measured from the top of the contour.

The **second command field** of the set (across coordinate) is the indent, in picas and points, as measured from the margin.

The coordinate values are ALWAYS measured from their relative starting points; top and margin.

Purpose:

To define a series of down and across coordinates to be used in setting a contour. The shape of the contour reflects the coordinates set up in the command string. The contour between each coordinate point is depicted as an imaginary straight line connecting the two points. Thus, the final contour shape will consist of a series of straight lines connecting each coordinate.

In order to compensate for the possibility of ascenders and descenders occurring in the text and to take into consideration the shape of the contour, the system will automatically go through a 3-step calculation in order to maximize the indent to be used on each line. The indent is calculated by the following steps:

1. The indent, along the contour, is first calculated at a point of 75% of the point size above the base line (i.e., at the ascender);
2. The indent, along the contour, is then calculated at a point of 20% of the point size below the base line (i.e., at the descender);
3. The indent of any down and across coordinates that fall in the range between steps 1 and 2.

Example:

Sample Input:

```
[cc22,1,8,10][wr0,2p6,7,4p,12,5p10,16,3p,21,11p]This text is being set at a measure
of 22 picas. The point size is 8 pt. and the leading is 10 pt. A wrap right command is
issued at the beginning of the first paragraph defining 5 sets of down and across
coordinates.[epThe first set defines a 0-pica depth (down coordinate) to be indented
on the right 2 picas and 6 points (across coordinate). The first line begins with a
proportionate amount of indent and gradually increases for every line until the
specified amount of 2 picas and 6 points is reached at a point to coincide with the
specified depth of 3 picas.[ep[cf2,12,14]The second set picks up where the first set
left off and defines another down coordinate of an additional 4-pica depth (7 picas
from the top) and an across coordinate of an additional 1 pica and 6 points indent (4
picas from the margin). Again, this 4-pica depth will be skewed an appropriate
amount on each line until the 1 pica and 6 points indent is exhausted to coincide
with the bottom of the 4-pica depth.[ep[cf1,10,12]The next set of coordinates
defines an additional 5-pica depth (12 picas from the top) and an additional 1 pica
and 10 points indent (5 pica and 10 points from the margin). After this 5-pica depth
is satisfied the remaining sets are executed after which the measure of 22 picas
will be resumed automatically.
```

Command Name: **Wrap Right (Contouring)—Continued**

Sample Output: (on 20 picas)

This text is being set at a measure of 22 picas. The point size is 8 pt. and the leading is 10 pt. A wrap right command is issued at the beginning of the first paragraph defining 5 sets of down and across coordinates.
The first set defines a 0-pica depth (down coordinate) to be indented on the right 2 picas and 6 points (across coordinate). The first line begins with a proportionate amount of indent and gradually increases for every line until the specified amount of 2 picas and 6 points is reached at a point to coincide with the specified depth of 3 picas.

The second set picks up where the first set left off and defines another down coordinate of an additional 4-pica depth (7 picas from the top) and an across coordinate of an additional 1 pica and 6 points indent (4 picas from the margin). Again, this 4-pica depth will be skewed an appropriate amount on each line until the 1 pica and 6 points indent is exhausted to coincide with the bottom of the 4-pica depth.

The next set of coordinates defines an additional 5-pica depth (12 picas from the top) and an additional 1 pica and 10 points indent (5 pica and 10 points from the margin). After this 5-pica depth is satisfied the remaining sets are execuated after which the measure of 22 picas will be resumed automatically.

Command Name:

Ragged Center

Input Code:	**Cancel Code:**
[rc	[xr Cancel Ragged Also [rl, [rr [te

Purpose:

Set the following type ragged center, and it **DOES NOT** automatically cancel hyphenation. This command is recommended to be used when setting centered heads.

Example:

Sample Input:

```
[rcThis will be ragged center until cancelled. As you can see this is a very valuable
command to have available. Hyphenation [cf3]IS NOT[cf1] automatically cancelled and
the ragged center will remain in effect until cancelled.[ep[xr
```

Sample Output: (on 20 picas)

This will be ragged center until cancelled. As you can see this is
a very valuable command to have available. Hyphenation **IS**
NOT automatically cancelled and the ragged center will remain
in effect until cancelled.

Computer Use:

The computer will set the paragraph by justification principles but will quad each line center. The last copy block terminator should be a [qc or [ep.

Caution:

1. [rc creates a justification mode of ragged center copy. [ep or [qc should be used as a copy block terminator at all times when using ragged mode.
2. If the copy block terminator is a [ql or single key quad left, all lines will center except last line which will quad left.

If the copy block terminator is a [qr or single key quad right the last line only will quad right.

If the copy block terminator is a [ep or a [qc the last line will be centered.

Command Name: Ragged Left

Input Code:	**Cancel Code:**
[rl	[xr
	Also [rr, [rc
	[te

Purpose:

Set the following type with a ragged left margin, and it **DOES NOT** automatically cancel hyphenation.

Example:

Sample Input:

```
[rlThis will be ragged left until a cancel ragged code is encountered in the tape
being processed. Ragged left is most frequently used for setting illustration
captions in magazine work. Hyphenation [cf3]IS NOT[cf1] automatically cancelled in
ragged mode.[ep[xr
```

Sample Output: (on 20 picas)

This will be ragged left until a cancel ragged code is encountered in the tape being processed. Ragged left is most frequently used for setting illustration captions in magazine work. Hyphenation **IS NOT** automatically cancelled in ragged mode.

Computer Use:

The computer will set the paragraph by justification principles but will quad each line right. The last copy block terminator should be a [qr or [ep.

Caution:

1. [rl creates a justification mode of ragged left copy. [ep or [qr should be used as a copy block terminator at all times when using ragged mode.
2. If the copy block terminator is a [ql the last line will be flush left.
   If the copy block terminator is a [qc the last line will be centered.
   If the copy block terminator is a [ep or a [qr the last line will flush right.

Command Name: Ragged Right

Input Code:	**Cancel Code:**
[rr	[xr Also [rl, [rc [te

Purpose:

Set the following type with a ragged right margin, and it **DOES NOT** automatically cancel hyphenation.

Example:

Sample Input:

```
[rrThis will be set as ragged right text with hyphenation until a cancel ragged
command is encountered by the system. The ragged style of setting text matter has
been in vogue for quite some time now.[ep[xr
```

Sample Output: (on 20 picas)

```
This will be set as ragged right text with hyphenation until a
cancel ragged command is encountered by the system. The rag-
ged style of setting text matter has been in vogue for quite
some time now.
```

Computer Use:

The computer will set the paragraph by justification principles but will quad each line left. The last copy block terminator should be [ep.

Caution:

1. [rr creates a justification mode of ragged right copy. [ep or [ql should be used as a copy block terminator at all times when using ragged mode.
2. If the copy block terminator is a [qc the last line will be centered.
   If the copy block terminator is a [ql or a [ep the last line will be flush left.
   If the copy block terminator is a [qr the last line will be flush right.

Command Name: **Conditional Hyphenation**

Input Code:	**Cancel Code:**
[ch	None
	[te

Example:

Sample Input:

```
Before the party ended, everyone congregated with much anti[chcipation around the
president of Penta Systems International, Inc., and pre[chsented him with a
pres[chent. After a long pause, the host smiled at his guests and he then thanked
each and every one of them for the pres[chent they had given him.[ep
```

Sample Output: (on 20 picas)

```
Before the party ended, everyone congregated with much anti-
cipation around the president of Penta Systems International, Inc.,
and presented him with a present. After a long pause, the host
smiled at his guests and he then thanked each and every one of
them for the present they had given him.
```

Computer Use:

This command is used to mark the point or points in words where they should be hyphenated, if necessary.

> **Caution:**
> This command **WILL NOT** override your system standards for the number of characters to fit on a line with a hyphen.

Command Name:

Allow Hyphenation/Cancel Hyphenation

Input Code:	**Cancel Code:**
[ah Allow Hyphenation	[xh Cancel Hyphenation
	[te

Purpose:

The system is such that each tape automatically allows hyphenation; therefore, this command is only used after an operator cancels hyphenation.

Example:

Sample Input:

```
[cc12]For this sample we will change the measure to 12 picas. Now is the time to set
manuscript in the hyphenated mode by hyphenating text to produce a full measure of
type utilizing the greatest number of characters on that line as possible.
Hyphenation is controlled by your system standards as to the minimum number of
characters to be left on a line and the minimum number of characters to be brought
down to the next line. We cancel hyphenation by the [xh command which will produce
justified lines in this example without the use of hyphenation. When operating in a
hyphenated mode you also control the number of hyphenated lines that may appear in a
row by using your system standards.[ep
```

Sample Output: (on 20 picas)

For this sample we will change the measure to 12 picas. Now is the time to set manuscript in the hyphenated mode by hyphenating text to produce a full measure of type utilizing the greatest number of characters on that line as possible. Hyphenation is controlled by your system standards as to the minimum number of characters to be left on a line and the minimum number of characters to be brought down to the next line. We cancel hyphenation by the command which will produce justified lines in this example without the use of hyphenation. When operating in a hyphenated mode you also control the number of hyphenated lines that may appear in a row by using your system standards.

Command Name: **Change Justifiers**

Input Code:	**Cancel Code:**
[cj—,—,—]	[nb
(See Description)	[te

Description:

The **first command field** specifies the minimum band size in units
The **second command field** specifies the optimum band size in units
The **third command field** specifies the maximum band size in units.

Purpose:

To allow an override to the font values for minimum, optimum and maximum spaceband values.

Example:

Sample Input:

```
[cj27,27,27]This type will be set using the spaceband values of 27 units minimum, 27
units optimum, and 27 units maximum. The next paragraph will change the band values
back to font recorded values by the normal band call.[ep[nbThis entire paragraph
will be justified using the font record band values which happen to be 10 units
minimum, 18 units optimum, and 27 units maximum. These unit values are based on a 54
unit base output module.[ep
```

Sample Output: (on 20 picas)

This type will be set using the spaceband values of 27 units minimum, 27 units optimum, and 27 units maximum. The next paragraph will change the band values back to font recorded values by the normal band call.

This entire paragraph will be justified using the font record band values which happen to be 10 units minimum, 18 units optimum, and 27 units maximum. These unit values are based on a 54 unit base output module.

Computer Use:

The computer uses these stated band values versus the font outlined band values and continues to follow the same justification logic as outlined at the beginning of this section.

Command Name: Change White Space

Input Code:	**Cancel Code:**
[cw__,__]	[nb
(See Description)	[te

Description:

The **first command field** specifies the size of the spaceband to allow automatic kerning of all characters The **second command field** specifies the size of the spaceband to allow letterspacing.

Purpose:

To allow you to use a new set of optimization values.

Example:

Sample Input:

```
[cw10,0]The justifiers (spacebands) have not been changed. The existing minimum band
value is 10 units, the optimum is 18 units and the maximum is 27 units. The spaceband
value of 10 has been set as an automatic kern point outlined in the first field. No
letterspacing has been outlined. The [nb cancels the [cw command as well as the [cj
command and returns to band values set in font record.[ep
```

Sample Output: (on 20 picas)

The justifiers (spacebands) have not been changed. The existing minimum band value is 10 units, the optimum is 18 units and the maximum is 27 units. The spaceband value of 10 has been set as an automatic kern point outlined in the first field. No letterspacing has been outlined. The [nb cancels the [cw command as well as the [cj command and returns to band values set in font record.

Computer Use:

When the specified band drive is ascertained for the kerning value, the system will automatically minus the line one unit per character in an attempt to bring up the line. When letterspace value is met, letterspacing takes effect. A zero in either field means you don't allow that function. When using this command, it will stay in effect until you use the [nb, then it will go back to font value.

Command Name: **Band Alert**

Input Code:	**Cancel Code:**
[ba___,___]	[ba0,0]
(See Description)	[te

Description:

The **first command field** is the minimum space band value you will allow, if less it will display on the PentaVue Terminal.

The **second command field** is the maximum space band value you will allow, if greater, it will display on the PentaVue Terminal.

Note: These two fields **MUST** be entered in tenths of a point.

Purpose:

To allow the user the ability to mark tight or loose lines so that they can be easily detected. They will be highlighted on the left when you put the tape thru the Flashable Mode and strike MAINT SPACE BAND with text on the screen.

Example:

Sample Input:

```
[ba145,350]In the following type any lines that are set with band drives less than
14-1/2 units or greater than 35 units will be highlighted on the PentaVue Terminal
when the band drives are displayed.
```

Sample Output: (on 20 picas)

Not Applicable.

Computer Use:

This command is working off the number in your Expansion Base set up in System Standards.

Caution:

This command has no effect on the justification of the tape, it is merely to highlight those lines on the PentaVue Terminal.

Command Name:

Increase/Decrease Character Width

Input Code:
[iw__] Increase Widths
[iw-__] Decrease Widths

Cancel Code:
[iw0], [ot__], [tr__] or
another [iw__] or [iw-__]
[te

Purpose:

To add to the current increase or decrease widths from the font of all characters following the command until another change character width command is encountered. Amount of increase or decrease is based on phototypesetter:

Values are;
VIP— 1/9 pt.— 2 digits max.
202— 1/54 em— 1 digit max.
APS— 1/10 pt.— 2 digits max.

Example:

Sample Input:

```
this is the end of normal widths.[ep[iw3]This is the beginning of increased character
widths. Note the change of three units to all letters (except the last letter of each
line) prior to another change character width command.[ep[iw0][el12]this is the end
of normal widths.[ep[iw-3]This is the beginning of decreased character widths. Note
the change of three units to all letters (except the last letter of each line) prior
to another change character width command.[ep[iw0]
```

Sample Output: (on 20 picas)

this is the end of normal widths.
This is the beginning of increased character widths. Note the change of three units to all letters (except the last letter of each line) prior to another change character width command.

this is the end of normal widths.
This is the beginning of decreased character widths. Note the change of three units to all letters (except the last letter of each line) prior to another change character width command.

Caution:
This command will stay in effect until changed or cancelled.

Command Name:

Track Command

Input Code:
[tr__]

Cancel Code:
Another [tr__] command
[te

Purpose:

To adjust all character widths and band values following the command according to predefined values in the font.

Example:

Sample Input:

```
This line is set with normal widths and bands.[ep[tr3]This line is set with the widths
and bands specified for track 3 of this font.[ep[iw0]This line is set with normal
widths and bands again.[ep
```

Sample Output: (on 20 picas)

This line is set with normal widths and bands.
This line is set with the widths and bands specified for track 3 of
this font.
This line is set with normal widths and bands again.

Computer Use:

When a track is requested in the tape, the computer will adjust all character widths and spaceband values according to the values defined for that track in the width table for the current font and point size.

Note: The commands Change Justifiers, Change White Space, and Normal Bands can be used to change the spaceband values without changing the width adjustment. The command Increase Widths can be used to add or subtract from the current width adjustment. The Override Track command can be used to completely override the width adjustment. Refer to the table on the next page which shows how these commands interact with the Track Command.

Caution:

1. Tight or very tight width adjustments can cause overlapping characters if used with kerning.
2. A tape with the command [tr0] will result in a command error. To return to track 0, use [iw-0] or [iw0].

Command Name: **Track Command—Continued**

COMMAND:	WIDTH ADJUSTMENTS	BAND VALUES
[tr X]	USE VALUES FROM TRACK 'X' OF THE FONT	USE VALUES FROM TRACK 'X' OF THE FONT
[ot X] [ot- X]	USE THE 'X' VALUE OF THE [ot COMMAND INSTEAD OF THE FONT	NOT EFFECTED
[iw X] [iw- X]	ADD 'X' VALUE OF THE [iw COMMAND TO THE CURRENT VALUE FROM THE FONT	NOT EFFECTED
[cj X, Y, Z] [cw X, Y]	NOT EFFECTED	USE 'X, Y, Z' VALUES OF THE [cj or [cw COMMAND INSTEAD OF THE FONT VALUES
[nb	NOT EFFECTED	USE VALUES FROM THE CURRENT TRACK OF THE FONT

Command Name: **Over-Ride Track**

Input Code:	**Cancel Code:**
[ot__]	[ot0] or another [tr__] command [te

Purpose:

The Over-ride Track command [ot__], cancels both the Track and the Increase/Decrease Width values that may be in effect when this command is keyed.

Example:

Sample Input:

```
I am in Track 2 and want to over-ride the value in that track, I merely key [ot0] and it
cancels the Increase/Decrease Width values and puts me back to normal.
```

Sample Output: (on 20 picas)

I am in Track 2 and want to over-ride the value in that track,
I merely key [ot0] and it cancels the Increase/Decrease Width
values and puts me back to normal.

Computer Use:

The value keyed in this command is absolute, it does not add or subtract to either the Tracks or the Increase/Decrease Width command, it over-rides them.

Caution:

The Increase/Decrease command will not cancel the Over-Ride Track command, it will add to or subtract from its values.

Command Name: Allow Kerning/Cancel Kerning

Input Code:	**Cancel Code:**
[ak Allow Kerning	[xk Cancel Kerning [te

Purpose:

To allow the kerning of previously specified character pairs in relative units per font record.

Example:

Sample Input:

```
This test is set without the use of kerning. PAY LESS FOR YOUR VACATION OR TRIP.
[ep[akThis test is set with the use of kerning. PAY LESS FOR YOUR VACATION OR TRIP.[ep
```

Sample Output: (on 20 picas)

This test is set without the use of kerning. PAY LESS FOR YOUR
VACATION OR TRIP.
This test is set with the use of kerning. PAY LESS FOR YOUR
VACATION OR TRIP.

Computer Use:

The computer uses this command to test for character combinations stored in the kerning file. It can be used with the [tr__], [iw-__] and [iw__] commands. The use of [xk will cancel the use of kerning. If kerning is allowed, the system will check the kerning switch in the current font to see if kerning is allowed in that font. If so, each character pair encountered is checked to see if it is stored as a kernable pair in the kerning file. The values of plus or minus width adjustment for the kernable pairs are stored in the font table of each face.

Caution:

For one tape or any part of a tape, unless specified as the standard in the kerning file.

Command Name: **White Space Fill**

Input Code:	**Cancel Code:**
[ws	[fj [ep [te

Purpose:

Justifies a line by inserting all leftover space at a point where [ws is keyed.

Example:

Sample Input:

```
Baltimore City[ws$37.05[fjBaltimore County[ws10.23[fjVioleteville 1973-
1974[ws37.05[epArbutus 1974[ws10.23[ep
```

Sample Output: (on 20 picas)

Baltimore City	$37.05
Baltimore County	10.23
Violeteville 1973–1974	37.05
Arbutus 1974	10.23

Computer Use:

The computer takes the text before [ws and quads it to the left and takes the text after the [ws and quads it right, then fills the remaining gap between the left and right with space.
Runover lines occuring after the line containing the [ws will automatically typeset flush right ragged left until the occurence of a copy block terminator.

Caution:

Only one [ws (White Space Fill Command) can be used in a line.

Command Name: **Leader Fill**

Input Code:	**Cancel Code:**
[lf	[fj [ep [te

Purpose:

This is used to set leaders within a line such as tabular work, table of contents and indexes, etc.

Example:

Sample Input:

```
Baltimore City[lf$37.05[fjBaltimore County[lf10.23[fjVioletville 1973-
1974[lf37.05[epArbutus 1974[lf10.23[ep
```

Sample Output: (on 20 picas)

Baltimore City $37.05
Baltimore County 10.23
Violetville 1973–1974.................................. 37.05
Arbutus 1974 .... 10.23

Computer Use:

The computer takes the text before [lf and quads it left and takes the text after the [lf and quads it right, then fills the middle with leaders. The computer automatically adds a space just before the leaders but proper space must be added after the [lf to align work.

A runover line after the leader fill function will automatically be flush right ragged left, regardless of whether you are in justified or ragged mode, until it reads a copy block terminator at which time the original justified or ragged mode will be reinstated.

Caution:

1. The [lf command will use the values from System Standards item numbers 87, 88, 89, 90.
2. Only one [lf command can be used in a line.
2. Only one [lf (Leader Fill Command) can be used in a line.

Command Name: **Variable Leader (for one line only)**

Input Code:
`[vl___,___,___,___]`

Cancel Code:
[fj
[ep
[te

Description:

The **first command field** is the PentaPal font character numeric position.
The **second command field** is the width of the character in points. You cannot make this field smaller than the width of the character in the PentaPal font.
The **third command field** is the minimum number of leaders you will allow. The computer will start a new line if necessary to give you the minimum number of leaders you requested.

Note: If text is to appear before and after the leaders, it is possible to have 1 less leader character than requested in this field due to leader alignment.

The **fourth command field** is the leader fill style number.
Style 1 — break the line wherever possible
Style 2 — break after leader fill
Style 3 — break before leader fill
Style 4 — break one word before leader fill
Style 5 — break in middle of leader fill

Purpose:

To specify a particular type of leader character for just one line and used in place of the Leader Fill ([lf) command.

Example:

Sample Input:

```
Center line em dash leaders[vl79]test[fjHyphen rule leaders[vl29]test[fjLower case
''x'' leaders[vl24]test[epNumber 8 leaders[vl73]test[ep
```

Sample Output: (on 20 picas)

Center line em dash leaders—————————————— test
hyphen rule leaders---test
Lower case "x" leaders xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx test
Number "8" leaders 888888888888888888888888888888888888 test

Computer Use:

The computer assigns the indicated leader fill character. Runover lines occuring after the [vl command will automatically be typeset flush right ragged left until the occurence of a copy block terminator.

Caution:
1. This command is only a temporary change for one line.
2. The use of certain characters may require the user to key the location of that character of the PentaPal font. All characters are dependent upon the user specifications and font layouts.
3. Only one type of leaderfill command can be used in a line.
4. If the last three command fields are not filled in it defaults to the width of the character as defined in the font and System Standards items 88, 89 and 90 for the width of the leader character, minimum number of leaders and style number.

FORMAT INTRODUCTION

There is ONLY one type of formats in PentaPal:

Universal
a. accessible by all jobs
b. maximum number is 9,999
c. none of the Universal formats are right protected

Command Name:

Define Universal Format/End Universal Format

Input Code:	**Cancel Code:**
[du__]	[ef
	Another [du__] of the same number
	[te

Purpose:

This command is used to define Universal formats by number. The maximum number of formats you can define is 9,999. It is always used when defining Universal formats by paper tape input or direct terminal input. It is not used if the Universal formats are being loaded from the format maintenance menu. (See Caution below)

Example:

Sample Input:

```
[du1][cc35,5,12,12][rr[ef
```

Sample Output: (on 20 picas)

Not Applicable.

Computer Use:

The computer will store everything between the [du command and the [ef command as the specified Universal format. All defined formats can be activated by [u__] or [bu__].

Caution:

1. When formats are originally defined via the Format Menu in Maintenance, it is understood that any update changes would also have to be made via the Format Menu.
2. When formats are defined via paper tape or Direct Input, it is feasible that they could exist in two places:
   a. in the input .TX file
   b. in the Format menu area

   (This occurs after the processing of the .TX file where the definitions originated. During this processing they are copied to the Format Menu area.)

   One of the 3 following methods must be used in order to maintain format update control:

   1) After initial JUSTIFY processing, delete all formats from the .TX input files; and make ALL update changes in the Format Menu area.

   2) Make ALL update changes in the .TX input files and NEVER use the Format Menu for this update purpose.

   3) Define ALL formats in a separate .TX file which can be sent through JUSTIFY process **ONLY**. This way any further update needed to the Universal formats can be made in the .TX file and run thru the JUSTIFY process **ONLY** again which will automatically update the format file (.JF).

Command Name: **Call Universal Format**

Input Code:
[u__]

Cancel Code:
The completion of the Universal format called or the call of another format.
[te

Purpose:

To activate a specified Universal format by number.

Sample Input:

```
[ul]Universal format one was called at the front of this paragraph. The format was
previously defined as a Universal Format, and will change the measure to 35 picas,
change to face 3, 12 point type and 12 point leading, ragged right.[ep
```

Sample Output: (on 20 picas)

Universal format one was called at the front of this paragraph. The format was previously defined as a Universal Format, and will change the measure to 35 picas, change to face 3, 12 point type and 12 point leading, ragged right.

Computer Use:

The [u__] command is used by the computer as an indicator to where the requested formats are to be placed.

Command Name: **Branch to Universal Format**

Input Code:
[bu__]

Cancel Code:
The completion of the branch format called
[u__]
[te

Purpose:

The Branch Command is used to temporarily exit from a working format, activate the branch format and upon its completion return to the first Universal format called.
Format 21 has been defined as the word one and the word three. Format 22 has been defined as spaceband two spaceband. To show the difference between a branch and a call format, each will be demonstrated below.

Sample Input:

```
[du21]one[bu22]three[ef[du22] two [ef[du23]one[u22][ef[u21][ep[u23][ep
```

Sample Output: (on 20 picas)

one two three
one two

Computer Use:

The computer will retain all copy merge entry points still unsatisfied and keep them in effect until the branch format is satisfied and then reinstate them accordingly.

Command Name: **Copy Merge**

Input Code:	**Cancel Code:**
[cm Copy Merge	Used once then discarded [u__] [te

Purpose:

Used in formats or tape to mark the spot where material from tape or formats is to be inserted. A format can contain many copy merges but for every copy merge in a format there must be a copy merge in the tape in order to function as devised or another format called to accomodate.

Example:

Sample Input:

```
[du26][el24][cf3,10,12][cm[qc[el24][cf4][cm[qc[el24][cf1,8,10][efThis is text then a 2
line head then more text.[ep[u26]line 1 of head[cmline 2 of head[cmNow we go back to
text[epThis is text then a 1 line head then more text[ep[u26]line 1 of head[cm[cmNow we
go back to text.[ep
```

Sample Output: (on 20 picas)

This is text then a 2 line head then more text.

line 1 of head

line 2 of head

Now we go back to text
This is text then a 1 line head then more text

line 1 of head

Now we go back to text.

Computer Use:

The Computer uses a copy merge as a door both to and from formats and tape.

Command Name: **Format at Leading**

Input Code:	**Cancel Code:**
[fl__,__]	When line leading count is satified.
(See Description)	[te

Description:

The **first command field** is the depth in picas, you wish to satisfy before calling a new Universal format. The **second command field** is the format number to be called.

Purpose:

To activate a specified Universal format by number after a certain depth in picas has been satisfied. This point in the copy would be unknown to the input keyboard operator.
The format at leading can be linked but **NOT** looped.

Example:

```
[du28][it0][ef–Format 28 cancels an indent tape
```

Sample Input:

```
[it5][cp8,10][fl2,28]This paragraph has a 5 pica indent on the left side. The format at
leading will call format 28 after 2 picas of depth has been satisfied. Format 28 will
cancel the indent. From that point on the copy will be justified to the full measure
with no indents.[ep
```

Sample Output: (on 20 picas)

This paragraph has a 5 pica indent on the left side. The format at leading will call format 28 after 2 picas of depth has been satisfied. Format 28 will cancel the indent. From that point on the copy will be justified to the full measure with no indents.

Caution:

1. The format used in the format at leading cannot have more than 256 characters. Characters from text that are merged into the format must be included in the 256 characters.
2. Be sure the format is loaded before using this command. You **CANNOT** change mode status, i.e. you cannot type in Secondary mode and then specify Normal mode in the format.
3. The format leading command works on the same principle as the vertical justification command. The total amount of picas in depth that can be used in this command is determined by the following formula: 32767 ÷ base = points ÷ 12 = total amount of picas

Command Name:
Define Tabular Column

Input Code:
[dt__,__,__, ... __]
(See description)

Cancel Code:
Another [dt command
[te

Description:

For VIP's this command can consist of up to 20 data fields (columns and/or gutters) containing 1 to 4 digits in picas and points, separated by a period or "p".

For CRT's this command can consist of up to 40 data fields (columns and/or gutters) containing 1 to 4 digits in picas and points, separated by a period or "p".

Both can carry a vertical alignment. Each column can have a different alignment. Vertical alignments are designated as b = bottom alignment in column, c = center alignment in column, t = top alignment in column, and g = gutter. Once defined, gutters are automatically skipped by the program.

For 202's and 8600's only, you can use a r = rule, which takes the place of the "g" so you will get vertical rules in the gutter.

If no alignment is specified, the program will use "t" top alignment.

Purpose:

To tell the computer the width (in picas and points) of each column and what vertical alignment each column is to be assigned.

Example:

Sample Input:

```
[dt5t,1g,6.6c,1r,6.6b][btThis will result in a 5 pica top alignment, then a 1 pica
gutter.[ep[nt6 picas and 6 points center aligned 2nd column.[ep[nt1 pica gutter with
a 1/2 point rule, 6 picas and 6 points bottom aligned third column.[ep[et
```

Sample Output: (on 20 picas)

```
This will result in                        | 1 pica gutter with a 1/
a 5 pica top         6 picas and 6 points  | 2 point rule, 6 picas
alignment, then      center aligned 2nd    | and 6 points bottom
a 1 pica gutter.     column.               | aligned third column.
```

Computer Use:

The above command defines three tabular columns with a gutter between the first and second columns, and a gutter with a rule between the second and third columns. This command will remain in effect for the entire tape unless changed. However, the type being set will only be in tabular if there is a begin tabular command, ([bt). Once the end tabular command ([et) is executed, the type will come out of tabular to be set in the last "cc" command in effect.

Caution:

1. When the measure of the tabular columns exceeds maximum measure minus 3 of the System Standards, the line numbers will not be output.
2. 300 justified lines between the [bt and the [nt command is the maximum when keying vertically.

Command Name: **Horizontal Tabular Setting**

Input Code:
[bt Begin Tabular mode
[nt go to Next Tabular column
[et End Tabular setting

Cancel Code:
[et cancels the tabular mode
[te

Description:
Each tabular unit must begin with a [bt. Each tabular column, after the first, is accessed by a [nt. The last tabular column is ended with a [et. Prior to these commands a define tabular column command ([dt) must have already been issued.

Purpose:
To allow tabular setting of copy.

Example:

Sample Input:

```
[dt5t,lg,6.6t,lg,6.6t][bt[rr[xhIn order to be able to set in the tabular mode you must
have a Begin Tabular command ([bt) which will take you to the first tab column.[epIn
order to go to the next tabular column you will need a Next Tab command ([nt) after
your last line terminator.[ep[ntNow I am setting in my next or second tab column. In
order to get to the next tab column you [cf3]MUST[cf1] key a Next Tab command after
your last line terminator.[ep[nt[xr[ahNow I am setting the last or third column of my
table. I order to get out of the tab mode I [cf3]MUST[cf1] key a End Tab command after
my last line terminaor.[ep[et[el6]The End Tabular command will take you out of the
tabular mode and allow you to set type as specified in your last ''cc'' command.[ep
```

Sample Output: (on 20 picas)

In order to be able to set in the tabular mode you must have a Begin Tabular command ([bt) which will take you to the first tab column.
In order to go to the next tabular column you will need a Next Tab command ([nt) after your last line terminator.

Now I am setting in my next or second tab column. In order to get to the next tab column you **MUST** key a Next Tab command after your last line terminator

Now I am setting the last or third column of my table. I order to get out of the tab mode I **MUST** key a End Tab command after my last line terminaor.

The End Tabular command will take you out of the tabular mode and allow you to set type as specified in your last "cc" command.

Command Name: **Horizontal Tabular Setting—Continued**

Computer Use:

The PentaPal tabular units are not restricted to one line across a page. Each tabular unit has both horizontal and vertical properties. The vertical depth of a tabular unit is determined by whichever column covers the largest vertical area. The remaining columns are then aligned at the top (t), bottom (b), or center (c), as designated by [dt.

Caution:

1. The vertical keyboarding of tabular lines is limited to 300 justified lines between the [bt and the next ([nt) next tab command.
2. The maximum number of next tab and define straddles that can be used in one line are dependent on the typesetter. (See Description of Define Tabular Column).

Command Name: Vertical Alignment

Input Code:	**Cancel Code:**
[at Align at Top	[nt
[ab Align at Bottom	[et
[ac Align at Center	[te

Description:

These are used after a begin tabular ([bt) or a next tabular ([nt) column command and are meant to be used to override the letters (specifiers) in the define tabular column measures [dt___,___, . . . ___] command. The only column **affected is that in which they are keyed.** Once the next [nt, [et or [bt commands are executed vertical alignment reverts back to that designated in the [dt command.

Purpose:

To override the vertical alignment, for one column only, from that originally designated in your [dt___,___,___, . . . ,___].

Example:

Sample Input:

```
[dt6c,lg,6t,lg,6b][rr[xh[cp7,8][btThis column is to have vertical alignment of center
on the deepest column.[ep[ntThis one will vertical align at the top on the deepest
column. To make sure this is the deepest column I am setting more text.[ep[ntThis one
will vertical align at the bottom of the deepest column.[ep[et[bt[atNow the customer
wants this column to vertical align at the top for this one time only. After I set the
begin tab command I also set a align at top command.[ep[ntThis column will also align
at the top.[ep[ntThis column will align at the bottom.[ep[et[btThis column will
default back to the define tab command vertical center.[ep[ntThis one top
align.[ep[ntThis one bottom align.[ep[et[xr[ah
```

Sample Output: (on 20 picas)

```
                        This one will vertical
This column is to       align at the top on
have vertical           the deepest column.
alignment of center     To make sure this is    This one will vertical
on the deepest          the deepest column I    align at the bottom of
column.                 am setting more text.   the deepest column.
Now the customer        This column will also
wants this column to    align at the top.
vertical align at the
top for this one time
only. After I set the
begin tab command I
also set a align at top                         This column will
command.                                        align at the bottom.
This column will        This one top align.
default back to the
define tab command                              This one bottom
vertical center.                                align.
```

Command Name: **Vertical Alignment—Continued**

Computer Use:

The computer uses the letters t, b, and c to align the respective columns of the table and uses the commands [at, [ab, and [ac to override for one column only, the position described in the tabular command.

Caution:

1. The COMMANDS affect only the column in which they are defined (but not the reoccurrence of that column), the LETTERS affect the entire tabular setting of each column.
2. The COMMANDS are in effect from point issued until encountering a [nt, [et or [bt, at which point the tabset position specifiers are used.

Command Name:

Define Straddle Head

Input Code:	**Cancel Code:**
[ds__]	[et [nt [te

Description:

The **command field** must be numeric and indicates the number of columns that are to be straddled. Do not count gutters when specifying how many columns you want to straddle. The define straddle head command [ds__] follows a [bt or [nt . The straddle head is then terminated by a next tab command [nt, which returns you to the column in which the define straddle head command was keyed. **Define staddle of one column is a legal command [ds1].**

Purpose:

To be able to set straddle heads in tabular matter. Straddle heads are words that go over 1 or more columns.

Example:

Sample Input:

```
[dt5t,lg,6.6t,lg,6.6t][bt[ds3]In order to set a straddle head you must have a Begin
Tabular mode command and a Define Straddle head command containing the number of
columns you would like to straddle[ep[ntIn order to be able to set in the first column
under the straddle head you must key a go to Next Tabular column.[ep[ntThen go to Next
Tabular column command will take you to the next tabular column, in this case it is
column two.[ep[ntThe next go to Next Tabular column command will take you to the next
column, in this case it is column three.[ep[et The End Tabular command will take you
out of the tabular mode and allow you to set type as specified in your last ''cc''
command.[ep
```

Sample Output: (on 20 picas)

In order to set a straddle head you must have a Begin Tabular mode command and Define Straddle head command containing the number of columns you would like to straddle[ep[ntIn order to be able to set in the first column under the straddle head you must key a go to Next Tabular column

In order to be able to set in the first tabular mode you must have a Begin Tabular mode command.

The go to Next Tabular column command will take you to the next tabular column, in this case it is column two.

The next go to Next Tabular column command will take you to the next column, in this case it is column three.

The End Tabular command will take you out of the tabular mode and allow you to set type as specified in your last "cc" command

Command Name:

Define Straddle Head—Continued

Computer Use:

Software is designed to automatically use gutter width(s) when it computes the measure necessary for the straddle width.

Caution:

1. You must be in tabular mode ([bt) in order to use a define straddle command ([ds__]).
2. When using the define straddle of one ([ds1]) be sure you key a [nt to bring you back to where you first keyed [ds1] in order to continue setting in that column.
3. When setting straddles and you end your straddle and you do not want to come back to where you first keyed the [ds__] you must key a [nt then your [et.

Command Name: Establish Underscore (Linotron 202, APS, 8600)

Input Code:
[eu___,___,___,___] Below Baseline
[eu-___,___,-___,___] Above Baseline

Cancel Code:
Another [eu___,___,___,___] command
[te

Description:

The **first command field** is the baseline deflection in points of the primary underscore. This field may be a negative number (use a hyphen) to set rule above the baseline.

The **second command field** is the weight of the primary rule.

1. APS in tenths of a point
2. 8600 in eights of a point
3. 202 based on 202 rule weights 0 to 9

The **third command field** is the baseline deflection in points of the secondary rule. This field may be a negative number (use a hyphen) to set rule above the baseline.

The **fouth command field** is the weight of the secondary rule.

Purpose:

To define the distance from the baseline to the underscores and their weights. If a define underscore command is not issued before usage of underscoring the default values will be used. The default for primary underscoring is two points below the baseline with a half point rule, and the secondary underscoring is four points below the baseline with a half point rule.

Example:

Sample Input:

```
[eu3,0,6,1]This sample input will set a [sul]primary underscore[xu 3 points below the
baseline with a half point rule. The [su2]secondary underscore [xu rule will be set 6
points below the baseline with a 1 point rule.[epThis sample was set on the Linotron
202 phototypesetter.[ep
```

Sample Output: (on 20 picas)

This sample input will set a primary underscore 3 points below the baseline with a half point rule. The secondary underscore rule will be set 6 points below the baseline ~~with a 1 point rule.~~
This sample was set on the Linotron 202 phototypesetter.

Caution:

If you are going from a primary rule to a secondary rule, you **MUST** end the primary rule before starting the secondary rule. Also, if you are going from a secondary rule to a primary rule, you **MUST** end the secondary rule before starting the primary rule.

Command Name:

Start/Cancel Underscoring

Input Code:
[su__] Start Underscore

Cancel Code:
[xu Cancel Underscore
[te

Description:

The **command fields** designates the type of underscore to be started, and can be a value from one to three.

1—Start primary underscore—which is the first two command fields in the Establish Underscore Command.
2—Start secondary underscore—which is the second two command fields in the Establish Underscore Command.
3—Start both underscores—which is all four fields of the Establish Underscore Command.

Purpose:

To designate the type of underscore, start point and end of an area to be underscored.

Example:

Sample Input:

```
Note: The [eu__,__,__,__] must be defined prior to using the [su__] command

[sul]This sample input[xu will set a primary underscore rule 2 points below the
baseline with a half point rule.[ep[su2]The secondary underscore[xu rule will be set
6 points below the base line with a 1 point rule.[ep[su3]Primary and secondary[xu
underscore used at same time.[ep[eu-7,.5,3,.5]Now when we call the [su3]it will
overscore and underscore at the same time[xu[ep
```

Sample Output: (on 20 picas)

This sample input will set a primary underscore rule 2 points
below the baseline with a half point rule.
The secondary underscore rule will be set 6 points below the base
~~line with a 1 point rule.~~Primary and secondary underscore used
at same time.
Now when we call the start underscore command it will overscore
and underscore at the same time.

Caution:

1. The cancel underscore command terminates any underscore which is in use at that time.
2. You ***MUST*** cancel the underscore you are in before you can call in another start underscore command.

Command Name: Rule Command (Linotron 202, APS, 8600)

Input Code:
`[ru__,__,__,__,__]`

Cancel Code:
[ep
[te

Description:

The below list of weights are for the Linotron 202 **ONLY**. For APS treat the **first command field** over 10 and for the 8600 treat the **first command field** over 8 for the weight of the rule.
The **first command field** specifies the Weight of the Rule.

0 = 1/2 point rule
1 = 1 point rule
2 = 1-1/2 point rule
3 = 2 point rule
4 = 3 point rule
5 = 6 point rule
6 = 8 point rule
7 = 12 point rule
8 = 16 point rule
9 = 24 point rule

The **second command field** specifies the Length of Rule. (If this field is 0 or is not present, the rule will be treated as a leader fill line.)
The **third command field** specifies the Tab Indications:

0—Does not extend into either gutters.
1—Extends into left gutter only.
2—Extends into right gutter only.
3—Extends into both gutters.

The **fourth command field** specifies the amount of space before the rule.
The **fifth command field** specifies the amount of space after rule.

Purpose:

You need only enter those fields which you are using.

Example:

Sample Input:

```
[rul][epWill fill line with weight 1 rule (all indents will be taken into
account.)[ep[rul,10][epWill draw a weight 1 rule ten picas long. (If less than 10
picas left in line, it will draw the rule to fit the
measure.)[ep[rul,10,0,.6,.6][epWill draw a weight 1 rule , ten picas long, with six
points space before the rule and six points space after the rule. (If less than 11
picas are available on line, the rule will be decreased to fit the measure. If there
is less than 1 pica on the line—the entire command is ignored.)[epThe rule below is 1
point in weight, 10 picas long, and ends with a [ep[rul,10][epThe rule below is 1 point
in weight, 10 picas long, and ends with a [qr[rul,10][qrThe rule below is 1 point in
weight, 10 picas long, and ends with a [qc[rul,10][qc[rul,10][ql[rul,10][fjThe rule
below is one point in weight and is indented 2 picas on each side with a [it[ep
[it2p,2p][rul][ep
```

Command Name:

Rule Command (Linotron 202, APS, 8600)—Continued

Sample Output: (on 20 picas)

Will fill line with weight 1 rule (all indents will be taken into account.)

Will draw a weight 1 rule ten picas long. (If less than 10 picas left in line, it will draw the rule to fit the measure.)

Will draw a weight 1 rule , ten picas long, with six points space before the rule and six points space after the rule. (If less than 11 picas are available on line, the rule will be decreased to fit the measure. If there is less than 1 pica on the line—the entire command is ignored.)
The rule below is 1 point in weight, 10 picas long, and ends with a [ep

The rule below is 1 point in weight, 10 picas long, and end with a [qr

The rule below is 1 point in weight, 10 picas long, and ends with a [qc

The rule below is one point in weight and is indented 2 picas on each side with a [it

Caution:

1. When using the [ru command to fill the line, that line **MUST** be ended with an [ep command.
2. If you are in a ragged center mode you **MUST** cancel the ragged before setting the rule.

INTRODUCTION TO VARIABLES

Before attempting to use the variables and variable math portions of the program, it is suggested that the user have a working knowledge of the overall system.

In the past, part of copy preparation was to specify column measure, indents, etc., in numeric values, to produce a given block of copy.

Presently, by utilizing variables, the Mark-Up person can instruct the computer how to calculate and use the required values from the copy being processed. The following pages will explain the variables in detail.

DEFINITION OF VARIABLES

Variables, in our usage, are amounts of space which are subject to change. There are two types of variables—*system variables* and *user variables*. During the INPUT H&J (JUSTIFY) of a tape certain variables can be examined and their values stored for future use. These are called *system variables*. There are 12 *system variable* available for your use. The following table gives the name and description of each *system variable*.

System Variable	Function
1s	distance to left margin
2s	distance to right margin
3s	current measure
4s	current font
5s	current point size
6s	current leading
7s	current left indent
8s	current right indent
9s	number of bands in line so far
10s	space assigned to bands
11s	space remaining in line
12s	current total lead in quarter points

There are 99 *user variables*. These are used for storing information and performing calculations within the system. These variables are identified by a number between 1 and 99 followed by the letter "v". All *user variables* are initialized to zero at the beginning of every tape.

Caution:
You cannot use a number higher than 99 in any of the *user variables* command fields.

Command Name: **Store Variables**

Input Code:
[sv__,__]

Cancel Code:
Another [sv command of the same user variable.
[te

Description:

The **first command field** specifies the variable position (1 to 100) that you wish to assign a *user variable.* The **second command field** specifies the *system variable* which indicates the function to derive the value of the *user variable* in the first command field. The second command field can be a *system variable* or a specified amount.

Purpose:

User variables may be entered anywhere in the tape. They are actually calculated during the INPUT H&J phase.

Example:

[sv1,2p]	this would store in *user variable* 1 a value of 2 picas.
[sv2,1s]	this would store in *user variable* 2 the value of the amount of space you have moved from the left margin.

The amounts stored in these user variables can be used by other commands. They may be used in commands such as hangs, indents, column measure, define tabs, etc.

You may store more than one variable in the same command by stringing out the amounts or functions you wish to assign. The variable position number that you specify is assigned the first value in the string and automatically incremented by one for each additional variable in the string.

[sv1,2p,1s,3s,2p6,4p,2s]

The above example would assign *user variable* 1 a value of 2 picas, *user variable* 2 a value of the current distance to the left margin, *user variable* 3 the current measure, *user variable* 4 is 2 picas 6 points, *user variable* 5 is 4 picas, *user variable* 6 is the current distance to the right margin. It uses consequtive *user variables* to store the values in these fields.

Sample Input:

```
Indent test line[sv1,1s][ep[it1v] until the indent is cancelled all text will be
indented a variable amount. (the width of the words Indent test line)[ep[it0]
```

Sample Output: (on 20 picas)

Indent test line
until the indent is cancelled all text will be indented
a variable amount. (the width of the words Indent
test line)

Command Name: **Variable Addition**

Input Code:
[va___,___,___]

Cancel Code:
Another [va command of the same *user variable.*
[sv command using same *user variable* stated in the 1st field.
[te

Description:

The **first command field** specifies the *user variable* where you want to put your answer.
The **second command field** specifies the *user variable* you want to add with the *user variable* in command field three.
The **third command field** specifies the *user variable* you want to add with the *user variable* in command field two.

The variable add command has 3 fields, each separated by a comma.

Purpose:

To add two user variables together you simply use the variable add command. The first command field defines the *user variable* which will hold the variable of the sum of the next two *user variable* fields. In other words you add the *user variable* in the second field to the *user variable* in the third field and place the answer in the *user variable* in the first field.

Example:

Mathematical Expression	PentaPal Coding
v1 = v2 + v3	[va1,2,3]

Sample Input:

```
This[sv2,1s][ep is the time.[sv3,1s][ep[va1,2,3][ep[oh1v]In this example variable one
will now contain the amount of space occupied by the words and spaces as shown-
''This is the time''. At the beginning of the paragraph we have issued an indent
command. The amount of the indent in 1v or the width of ''This is the time.''[ep
```

Sample Output: (on 20 picas)

This
is the time.

In this example variable one will now contain the amount of space
occupied by the words and spaces as shown—
"This is the time". At the beginning of the paragraph
we have issued an indent command. The amount
of the indent in 1v or the width of "This is the
time"

Computer Use:

The system will always use optimum space band values when measuring the width of the space bands used in that line.

Command Name: Variable Add W/Zero

Input Code:	**Cancel Code:**
[va0,___,___]	Another [va0, command using variable **position 29**. [sv command using *user variable* **position 29**. [te

Description:

The 0 in the **first command field** specifies that the variable add has a new function. This function is to add the values of all *user variable* positions starting with the specified variable in the second field.
The **second command field** specifies the first *user variable* and the **third command field** specifies the last *user variable* to be added together. The sum is stored in *user variable* **number 29** (selected by PentaPal automatically).

Purpose:

The answer will always go in *user variable* **position 29**.

Example:

Mathematical Expression	PentaPal Coding
v29 = v3 + v4 + v5 + v6 + v7 + v8	[va0,3,8]

The above coding means to add the values in *user variable* 3, *user variable* 4, *user variable* 5, *user variable* 6, *user variable* 7, *user variable* 8, and put the sum in variable **position 29**.

Command Name: Variable Subtraction

Input Code:
[vs___,___,___]

Cancel Code:
Another [vs command of the same *user variable*.
[sv command of the same *user variable* stated in the first command field.
[te

Description:

The **first command field** specifies the *user variable* where the answer is placed.
The **second command field** specifies the *user variable* to subtract from.
The **third command field** specifies the *user variable* to subtract by.

To substract two variables you simply use the *system variable* subtract command which takes the *user variable* in the third field and subtracts it from the *user variable* in the second field and places the result in the *user variable* in the first field.

Example:

Mathematical Expression	PentaPal Coding
v1 = v2 − v3	[vs1,2,3]

Sample Input:

```
TT[sv3,1s][epTTTT[sv2,1s][epUser variable 3 has been defined as the width of 2 cap
T's. User variable 2 has been defined as the width of 4 Cap T's. Using the system
variable substraction command, user variable 3 is subtracted from user variable 2
and the answer is stored in user variable 1.[ep [vs1,2,3][ih1v]At the front of this
paragraph a hang command of 1v was issued. This is the difference of user variables 2
and 3 or the width of 2 Cap T's.[ep[ih0]
```

Sample Output: (on 20 picas)

TT
TTTT
User variable 3 has been defined as the width of 2 cap T's. User
variable 2 has been defined as the width of 4 Cap T's. Using the
system variable substraction command, *user variable* 3 is subtracted
from *user variable* 2 and the answer is stored in *user variable* 1.
At the front of this paragraph a hang command of 1v was issued.
  This is the difference of user variables 2 and 3 or the width of
  2 Cap T's.

Caution:
The largest number **MUST** be in the second command field. If not, you will zero yourself out and come up with an Error.

Command Name:

Variable Subtraction W/Zero

Input Code:	Cancel Code:
[vs0,__,__]	Another [vs0 command using *user variable* **position 29**. [sv command using *user variable* **position 29**. [te

Description:

The 0 in the **first command field** specifies that the *system variable* subtract has a new function. This function is to compare the contents of the next two *user variable* fields.

The **second and third command field** specify the *user variables* to be compared. If the content of the *user variable* in the second field is less than the content of the *user variable* in the third field, then the content of the *user variable* in the second field is placed in *user variable* **number 29** (selected by PentaPal automatically).

Example:

Mathematical Expression	PentaPal Coding
$v29 = v7$ if $v7 < v4$	[vs0,7,4]

The above coding means **if** the content of *user variable* seven is less than the content of *user variable* four, place the value of *user variable* seven into variable **number 29.**

Caution:

If the content of *user variable* 4 is less than the content of *user variable* 7, you will zero yourself out and come up with an Error.

Command Name: **Variable Times (Multiplication)**

Input Code:
[vt__,__,__]

Cancel Code:
Another [vt command of the same *user variable.*
[sv command using the same *user variable* stated in the first command field.
[te

Description:
The **first command field** specifies the *user variable* where the answer is placed.
The **second command field** specifies a *user variable* to be multiplied.
The **third command field** specifies a *user variable* by which to multiply the second command field.

To multiply the content of two variables, use the *system variable* multiply command, which takes the content of the *user variable* in the second field and multiples it by the content of *user variable* in the third field, and places the product in the *user variable* of the first field.

Example:

Mathematical Expression	PentaPal Coding
v1 = v2 × v5	[vt1,2,5]

The above coding means that the content of *user variable* two is multipled by the content of *user variable* five and the result is placed in *user variable* one.

Command Name: Variable Times W/Zero

Input Code:	**Cancel Code:**
[vt0,__,__]	Another [sv command of using the same *user variable* as command field 3. [te

Description:

The 0 in the **first command field** means that *system variable* multiply has a new function. This function is to treat the **second command field** as an absolute number, and multiply the value of the **third command field** by this number and place the result in the third field, (which REPLACES the original value stored in this *user variable*).

Example:

Mathematical Expression	PentaPal Coding
v1 = 4 × v1	[vt0,4,1]

The above coding indicates that the value stored in *user variable* one has been multiplied by the absolute number four, and the product has been placed back in *user variable* one. The new value in *user variable* one REPLACES the original content.

Command Name: Variable Division

Input Code:	Cancel Code:
[vd__,__,__]	Another [vd command of the same *user variable*. [sv command using the same *user variable* position stated in the first command field. [te

Description:

In order to divide two variables, use the *system variable* divide command, which takes the *user variable* in the **second command field** and divides it by the *user variable* in the **third command field**, and places the dividend in the *user variable* in the **first command field.**

Example:

Mathematical Expression	PentaPal Coding
v1 = v6 ÷ v7	[vd1,6,7]

The above coding means to take the content of *user variable* seven and divide it into the content of *user variable* six, and place the answer into *user variable* one.

Command Name: Variable Division W/Zero

Input Code:
[vd0,__,__]

Cancel Code:
Another [sv command using same *user variable* as command field 3.
[te

Description:

The 0 in the **first command field** specifies that *system variable* divide has a new function. This function is to divide the *user variable* in the **third command field** by the absolute number in the **second command field** and place the results back in *user variable* in the third field.

Example:

Mathematical Expression	PentaPal Coding
v7 = v7 ÷ 2	[vd0,2,7]

The above coding indicates the division of the value of variable seven by the absolute number 2, with the answer being returned to *user variable* seven.

Command Name: Variable Greater Than

Input Code:
[vg___,___,___]

Cancel Code:
Another [vg command of the same *user variable* position.
[sv command of the same *user variable* as command field 1.
[te

Description:
To determine which variable contains the greater value, by comparing the *user variable* in the **second command field** with the *user variable* in the **third command field**. The result of the comparison is to place the greater value into the *user variable* defined in the **first command field**.

Example:

Mathematical Expression	PentaPal Coding
v1 = greater of v14 or v41	[vg1,14,41]

Sample Input:

```
T[sv41,1s][epTT[sv14,1s][epUser variable 41 has been defined as the width of 1 Cap T
and user variable 14 as the width of 2 Cap T's. Using the System Variable Greater
Than command the 2 user variables are compared and the greater of the 2 is placed in
user variable 1.[vg1,14,41][ep[ih1v]At the front of this paragraph a hang command of
1v was issued. This is the greater of the 2 user variables or the width of 2 cap T's.
This particular command is helpful in configuring longest lines of poems to create
the proper indent to set the poem flush left on the indent.[ep[ih0]
```

Sample Output: (on 20 picas)

T

TT

User variable 41 has been defined as the width of 1 Cap T and user variable 14 as the width of 2 Cap T's. Using the System Variable Greater Than command the 2 user variables are compared and the greater of the 2 is placed in *user variable* 1.

At the front of this paragraph a hang command of 1v was issued. This is the greater of the 2 user variables or the width of 2 cap T's. This particular command is helpful in configuring longest lines of poems to create the proper indent to set the poem flush left on the indent.

Command Name:

Variable Greater W/Zero

Input Code:	**Cancel Code:**
[vg0,__,__]	None [te

Description:

The function of this command is to round up to the nearest point those values contained in the defined *user variable* expressed in the **second command field**, through those in the **third command field**. The new values will be retained in the original *user variable*.

Example:

Mathematical Expression	PentaPal Coding
round up to nearest point v4 thru v9	[vg0,4,9]

The above coding means *user variables* 4 *through* 9 will be rounded up to the nearest point.

Command Name: Variable Less Than

Input Code:
[vn__,__,__]

Cancel Code:
Another [vn command of the same *user variable* positions.
[sv command of the same *user variable* as command field 1.
[te

Description:

To find the lesser of two variables, by comparing the *user variable* in the **second command field** against the *user variable* in the **third command field**, and placing the lesser value into the *user variable* defined in the **first command field**.

Example:

Mathematical Expression	PentaPal Coding
v1 = lesser of v2 or v3	[vn1,2,3]

Sample Input:

```
T[sv3,1s][epTT[sv2,1s][epUser variable 3 has been defined as the width of 1 Cap T and
user variable 2 as the width of 2 Cap T's. Using the System Variable Lesser Than
command, the two user variables are compared and the lesser of the two is placed in
user variable 1.[vn1,2,3][ep[ih1v]At the front of this paragraph a hang command of 1v
was issued. This is the lesser of the 2 user variables or the width of 1 Cap T.[ep[ih0]
```

Sample Output: (on 20 picas)

T
TT
User variable 3 has been defined as the width of 1 Cap T and user variable 2 as the width of 2 Cap T's. Using the System Variable Lesser Than command, the two user variables are compared and the lesser of the two is placed in user variable 1.
At the front of this paragraph a hang command of 1v was issued.
 This is the lesser of the 2 user variables or the width of 1 Cap
 T.

Command Name: **Variable Less Than W/Zero**

Input Code:	**Cancel Code:**
[vn0,__,__]	None [te

Description:

The purpose of this command is to allow a means to round down to the nearest point those values contained within the *user variables* in the **second command field** and the **third command field**. These new values are retained in the originally defined *user variable* fields.

Example:

Mathematical Expression	PentaPal Coding
round down to nearest point v4 thru v9	[vn0,4,9]

The above coding means *user variable* 4 through *user variable* 9 will be rounded down to the nearest point.

Printing the Variables

In order to print the values in the *user variable* you simply type [vg0,0,0] at the beginning of your tape and any time a *system variable* command is called the values will print on the line printer in points.